Secure Mobile Device Integration for Automotive Telematics

By

Beniamino Piero Bruno, BComp

A dissertation submitted to the
School of Computing
in partial fulfilment of the requirements for the degree of

Bachelor of Computing with Honours

University of Tasmania

November, 2005

Declaration

This thesis contains no material which has been accepted for the award of any other degree or diploma in any tertiary institution, and that, to my knowledge and belief, the thesis contains no material previously published or written by another person except where due reference is made in the text of the thesis.

…… ...…….

Beniamino Piero Bruno Date

Abstract

The vehicle is a challenging environment in which to interact with computing devices. Therefore, the vehicle environment offers computing a unique challenge, in that a method for safe and secure mobile device integration is required in order to provide a suitable communications channel for interaction with devices without distracting from the primary driving task.

Moreover, a security architecture is required for mobile device integration in the vehicle paradigm. This architecture must be scalable, efficient and most importantly built on trusted and mathematically sound algorithms.

This thesis examines the relevant literature in the field of automotive telematics, including the notion of mobile device integration. Moreover, issues in the vehicle paradigm are also discussed which include driver distraction, and the legal ramifications of in-vehicle mobile phone use. From a system design view point this thesis will then provide an overview of the design requirements for telematics products, and outline possible security protocols which could be implemented on constrained mobile devices.

The ultimate aim of this thesis is to develop a security architecture for mobile device integration for automotive telematics based on the simple network management protocol (SNMPv3) user security model.

Acknowledgments

Firstly I wish to thank my supervisors Dr. Daniel Rolf and Dr. Waheed Hugrass. I thank you both for your guidance, especially at times of extreme stress.

I also wish to thank Jacky Hartnett for her guidance with the security side of this project, as even though I was not one of her students she was always happy to provide assistance.

I also wish to thank my family for supporting me in this effort. I am sorry for the grey hair I have given you all. To my brother thank you for proofing my work even though you were extremely busy. To my mother, thank you for being the best taxi driver in town. And thankyou for only grumbling quietly when you had to get out of be to take me into uni in the middle of the night.

To the honours lads, thankyou for being the bunch of amusing geeks that you all are at heart. It's always good to take a break at times of stress and go for a kick of the footy. To the boys in honours room 3 now affectionately known as the *"Furnace of Productivity"* I think we did well considering we were locked in a broom closet all year.

To my mates outside of university I thank you all for providing me with a link to the outside world and a way to put things in to perspective. There is nothing like relaxing after a hard day with a few mates and some cool aperitifs.

And finally on a serious note I wish to dedicate this thesis to the memory of Ben Lockhart who passed away the day before this work was due (November 1 2005). You will be missed mate.

Table of Contents

Listing of Figures and Tables

Figures

Tables

1

1. Introduction

Currently, mobile telecommunications technology is undergoing an evolutionary back flip. Modern mobile telecommunications devices are the direct descendents of the in-built car phones of the 1940s. However the devices of today are heading back to the vehicle.

From their humble beginnings as a the device of the businessman, modern mobile phones are now the 'must have' device for all. In fact, Fortunati (2001) shows that it was the extensive use of mobile phones in the workplace that 'dragged' the mobile phone into the domestic environment, and transformed the device into a 'personal technology' that can seamlessly follow the user from the workplace to the home.

As the popularity of the mobile phone has increased, so has the notion of the 'mobile worker'. The need to work from anywhere has taken the mobile phone out of the office and into other environments, including the vehicle.

Recent research estimates that worldwide mobile phone adoption will reach two billion by 2007 (instat.com 2003). If this figure is coupled with other estimates that suggest that 85 percent of mobile phone owners use their mobile phone at least occasionally while driving (Goodman et al. 1997), by 2007 there could be approximately 1.7 billion drivers worldwide who are likely to use their mobile phone at some point while in control of a vehicle. Moreover, Hahn et al. (2000) estimate that mobile phone users spend 60 percent of their total mobile conversation time conversing while in control of a vehicle.

It is safe to predict that with the worldwide growth of demand for mobile services, and with the increasing notion of the ‘mobile worker’, mobile phone users will expect more from their device. Moreover, mobile phone users will look to their vehicles as not just a mode of transportation, but a mobile work environment. This will lead to increased use of mobile phones in vehicles. As such, in order for mobile devices to be successfully incorporated into vehicles they require a method for safe vehicle integration, which employs advanced human interaction techniques, an architecture for authentication, while requiring limited cognitive demand for operation.

1.1. Thesis Aims

The following are the goals which were examined in this thesis:

1. Review the relevant literature in the field of automotive telematics, and mobile device integration.
2. Explore the issues in developing systems for the vehicle environment. Including human computer interaction requirements, and the safety implications of in-vehicle mobile phone use.
3. Review relevant literature relating to the background of security methodologies.
4. Outline a security architecture for use on mobile devices of differing specifications. Where the underlying algorithms which comprise this architecture are both proven and mathematically sound.
5. Extend this system into a protocol for use in a networked environment.
6. Evaluate the Ang face recognition system for use as a possible user authentication method.

2

2. Literature Review

The following chapter will examine the existing literature relevant to in-car telematics products, their design and their focus on security. This chapter will also explore a range of issues including the legal implications of in-vehicle mobile phone use and the issue of driver distraction. Moreover, the principles of human-computer interaction will be introduced, as this forms the base design requirements for in-car mobile device integration and telematics products. Finally, traditional security mechanisms will be discussed, with particular focus on the rationale for, and suitability of, their application in the mobile environment, concentrating on methodologies for transparent and limited interaction user and device authentication.

2.1. Background

Automotive electronics have developed substantially since Paul Galvin the founder of Motorola developed the first car radio in the 1930s (Motorola 2005). Today, the scope of automotive electronics has evolved to include everything from entertainment systems, to monitored fleet services, and navigation systems. This new generation of advanced automotive electronics is known as telematics. The average modern vehicle contains approximately twenty computers. These systems are largely ubiquitous, and include functions ranging from the compact disk player in the center console, to the digital displays mounted in the dashboard, and the anti-lock breaking system (ABS), traction control, and fuel injection systems in the motor and associated systems.

2.1.1. Telematics

The term ‘telematics’ was derived from a translation of the term ‘télématique’, which first appeared in a historically significant report entitled *L'informatisation de*

la Société (Nora and Minc 1968) (translated: The Computerisation of Society) presented to the President of France in 1968. In this report the term 'telematics' was used to define the merger of telecommunications and computer technology. This definition still holds true today. However, modern day telematics engineering is focused on merging personal telematics devices such as mobile phones, with other telematics genres in an attempt to integrate telematics into the vehicle and other contexts.

2.1.2. Ubiquitous and Pervasive Computing

The notion of 'ubiquitous' and 'pervasive' computing was first introduced by the visionary researcher Weiser (1999) who states *"the most profound technologies are those that disappear. They weave themselves into the fabric of everyday life until they are indistinguishable from it."* This notion is realised by embedding computers throughout an environment. These invisible embedded devices are often connected in a redundant full mesh topology, where they communicate the status of the environment. This allows the overall system to appear 'smart' as the environment itself can sense and respond to changes. These systems are ubiquitous as they are unobtrusive to the user and therefore operate independently without user interaction.

2.2. The Automotive Telematics Revolution

A report by the principle analyst of the Telematics Research Group states that the automobile has undergone a vast transformation over the past two decades, shifting from an analogue machine consisting of predominantly mechanical control systems, to a digital car containing mostly computer-based control systems (Juliussen 2003). The notion of the 'digital car' is an automobile, which contains multiple dedicated and interconnected computing devices, which together create both a telematics and vehicle control system.

Traditionally telematics systems consist of three basic capabilities (Juliussen 2003). These consist of one or more two-way communication pathways, which can include wireless networking mediums such as Bluetooth or 802.11 variants, for interconnection with other devices in the vehicle, to mobile voice/data based networks including GSM, CDMA and GPRS, EV-DO to provide internet connectivity, or a medium to enable real-time services.

The second key capability of an automotive telematics system is a global positioning system that can be used to provide location based and fleet monitoring services. Finally, a computing platform is required for system control and an interface to automotive electronics systems, including system buses and input/output devices such as inbuilt vehicle displays.

This is however a traditional overview of the requirements of a telematics system. As stated in Section 2.1.1 there has been a shift in the telematics industry to provide means for the integration of personal telematics devices in the vehicle.

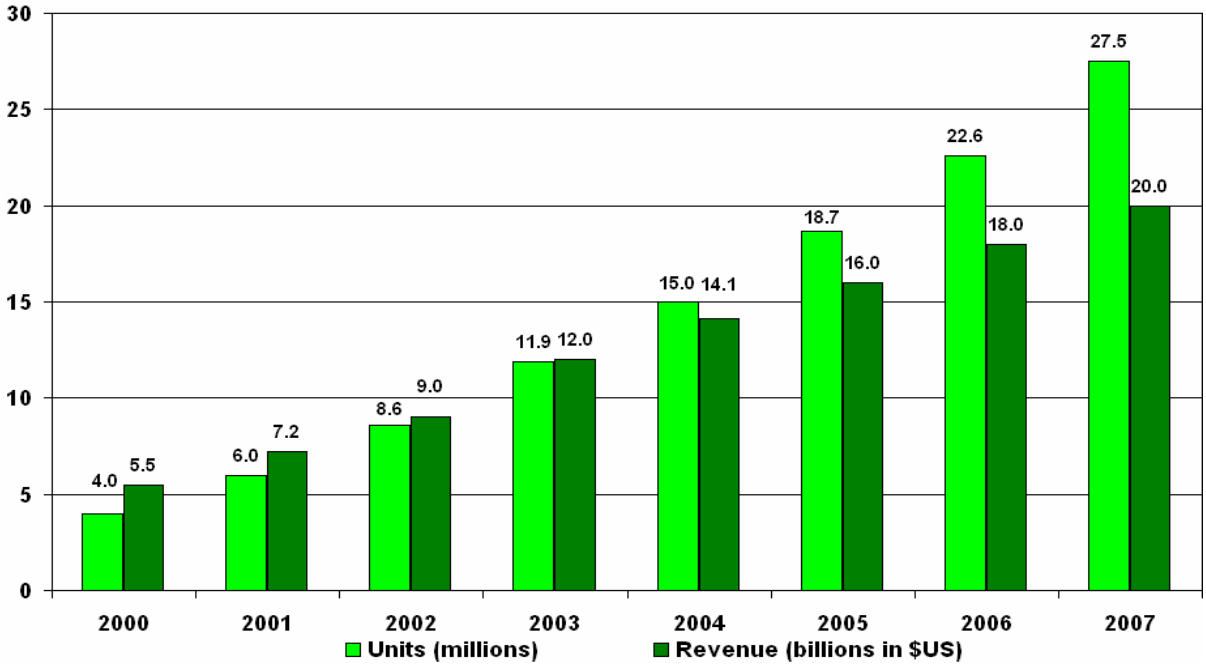


Figure 2.1. Worldwide Telematics Forecast (Strategy Analytics)[1]

Figure 2.1 above illustrates market research estimates of the prospected growth of the telematics industry market by 2007. These projected figures include both original-equipment-manufacture (OEM) and aftermarket units. It is also estimated that by 2007, approximately 55 percent of new vehicles sold worldwide will be telematics enabled, compared to just 7.5 percent in 2000 (Zhao 2002).

[1] Strategy Analytics - http://www.strategyanalytics.net/

2.2.1. What is the 'killer application'?

It is difficult to estimate that there will be a single application that will be the driving force of the telematics industry. The primary reason for this is that the requirements for user services differ in different regions of the world.

Table 2.1. Telematics Market and Technology Trends by Region (2004)[2]

	Factors	Telematics Trends	Comments
USA	75 people/sq mile 60 autos/sq mile 53% mobile phone use	Safety & Security TM Mobile device TM growing Navigation TM emerging	GM telematics bundling BMW & M-B strong What will Ford do?
Japan	872 people/sq mile 501 auto/sq mile 65% mobile phone use	Navigation TM dominant VICS traffic information Rapid growth projected	Toyota G-Book as standard? Nissan Carwings as standard? Honda InterNavi
Germany	598 people/sq mile 343 autos/sq mile 76% mobile phone use	Mobile device TM is leader Navigation TM important TMC traffic information	BMW & M-B home market Hands-free mobile phone law Germany is European leader
Western Europe	175 people/sq mile 74 autos/sq mile 83% mobile phone use	Mobile device TM is leader Some navigation TM Some safety/security TM	Hands-free mobile phone law OEM home markets: France, Italy, Sweden
Other Regions	Australia & NZ S. Korea Other countries	Holden & luxury autos TM from all Korean OEMs Primarily in luxury autos	Australia: Holden home market Korea: 3G mobile phone leader Mostly European luxury cars

Table 2.1 above shows how the focus of the telematics industry changes according to the region (Telematics Research Group 2004). America is primarily concerned with safety and security telematics, due to a relatively low mobile phone adoption rate and an average of only 60 vehicles per square mile. However mobile device integration is an area of growth in the region. Asia is focused on providing advanced navigation systems, as they have overpopulated motorways with an average of 501 vehicles per square mile. Finally, countries in the European Union are more dedicated toward the development of services for mobile device integration in vehicles, due to the high adoption rate of mobile phones in the region and the introduction of laws, which regulate mobile phone use in vehicles.

2.3. Legislation

The use of mobile phones while driving has been shown to contribute to traffic accidents. Redelmeier and Tibshirani (1997) found that the likelihood of a motor

[2] Telematics Research Group (Europe) - http://www.telematicsresearch.de/

vehicle collision was increased four-fold when using a mobile phone while in control of a vehicle (Figure 2.2).

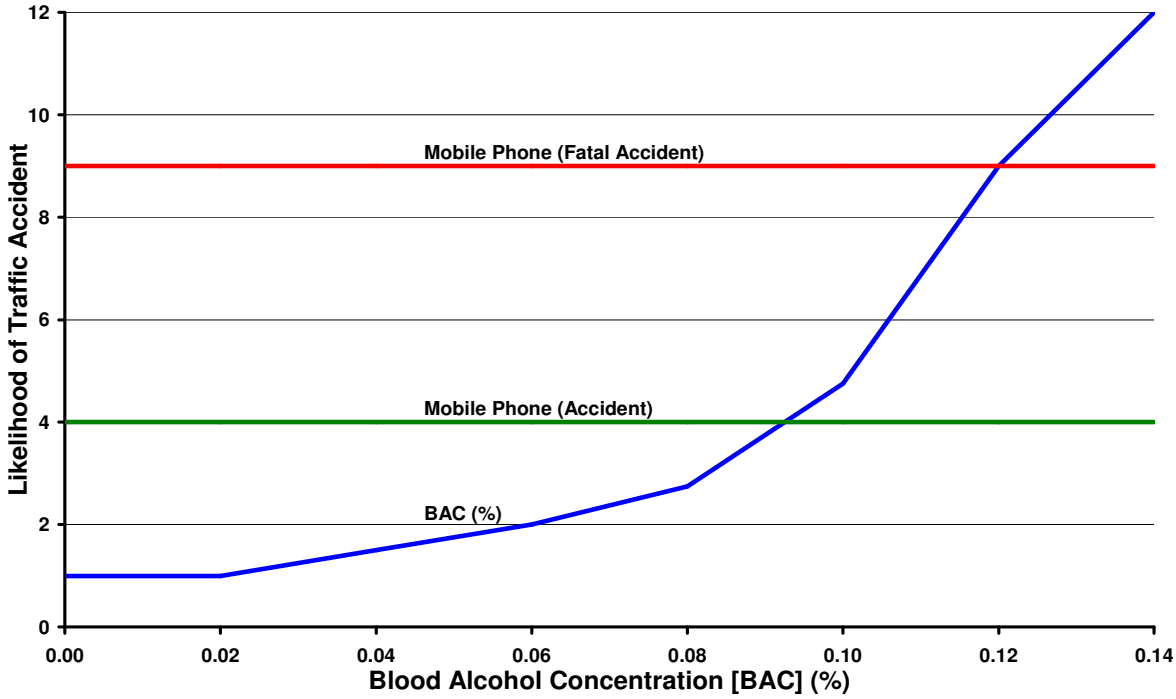


Figure 2.2. Effect of Alcohol and Mobile Phone use on driving ability

The above Figure is put in perspective when compared with a two-fold risk increase in the likelihood of an accident while driving a vehicle with a blood alcohol level of 0.06 percent, which is 0.01 percent higher than the Australia legal limit (Brick 1996). Moreover, a recent study concluded that the risk of a fatal vehicle accident increases nine-fold with the use of a mobile phone (Violanti 1999). It is due to findings such as these that an increasing number of countries and jurisdictions around the world are enacting legislation to limit the use of mobile phones while in control of a motor vehicle.

2.3.1. Australian Legislation

Australian legislation relating to in-vehicle mobile phone use is incorporated into the state and territory traffic regulations. In Tasmania as in other jurisdictions of Australia the use of hand-held mobile phones is banned while the vehicle is moving or stationary but not parked (State Government of Tasmania 1999). It should be noted that under this legislation hand-held two-way radios do not fall under the

definition of 'hand-held mobile phone'. Moreover, this legislation does allow the use of mobile phones used in a hands-free capacity.

2.3.2. American Legislation

Legislation in countries such as the United States of America is dramatically less stringent than that of Australia and other regions. In the USA mobile phone use in vehicles is also legislated at the state level of government (National Conference of State Legislatures 2003). Other than New York, Massachusetts has the most stringent legislation where mobile phone use is permitted as long as it does not interfere with the operation of the vehicle and one hand remains on the steering wheel at all times. Legislation in the states of Kentucky, Louisiana, Mississippi, Nevada, Oklahoma and Oregon prohibit local jurisdictions from restricting the use of mobile phones while driving. Finally in the states Maine and New Jersey drivers below the age of 21 are prohibited from using a mobile phone while driving, where drivers above the age of 21 are unrestricted.

2.3.3. Penalties

Another aspect of the current legislation, although primarily in America, is that the penalties for non-conformation are not sufficient to deter offenders. Brooklyn, Ohio was the first American jurisdiction to enact legislation that bans in-car mobile phone use. However, the penalty for non-conformation is only US$3 (Hahn, Tetlock and Burnett 2000). This can be compared to Australian law where the highest penalty occurs in the state of New South Wales where the fine for non conformation is AU$226 and 3 demerit points (Queensland Business Review 2003).

Adequate Legislation?

It is interesting to note that the only country that has proof of the effectiveness of their adopted legislation is also the country that enforces the harshest penalty.

Since November 1 1999 the use of a portable hand-held telephone device while in control of a vehicle was prohibited in Japan, unless the vehicle is stationary, or it is an emergency. The penalty for non-conformation is up to three months in prison or

fines of up to 50,000 Yen (AUD$477[3]). Moreover, just 12 months after the legislation was enacted there was a 52.3 percent decline in traffic accidents where the driver was using a mobile phone, a 53.3 percent drop in the number of injuries from accidents where the driver was using a mobile phone, and a 20 percent decrease in the number of fatalities from traffic accidents where a mobile phone was used by the driver (Royal Society for the Prevention of Accidents (RoSPA) 2001).

Williams (2002) states that the current enacted legislation does not directly address the problem of mobile phone use while in control of a motor vehicle, and therefore these laws are likely to have only a limited effect. It can be concluded that the current legislation does not successfully address the problem because the punishment occurs after the fact, if at all. Therefore, drivers are not likely to adhere to the restrictions.

2.4. Cognitive Distraction

It is interesting to note that the legislation which bans the use of hand-held mobile phones while driving allows the use of these devices in a hands free capacity. The aforementioned legislation makes the assumption that any interference from mobile phone use while in control of a motor vehicle is related to peripheral factors, which include dialling and holding the phone while conversing. Redelmeier and Tibshirani (1997) have shown that mobile devices which offer hands-free operation offer no safety advantages compared to hand-held devices.

Many studies attribute the increased risk of vehicle accidents to a lack of attention on the primary driving task while conversing on the mobile phone. Strayer and Johnston (2001) conducted studies on the effect of single operation tasks such as attending to auditory input in the form of listening to the car radio or a recorded audio book, and concluded that single operation tasks are not sufficient to produce an impairment in driving performance.

[3] Exchange rate calculated at http://www.x-rates.com/calculator.html (accessed 20 Oct 2005)

2.4.1. Inattention Blindness

Strayer and Johnston et al. (2003) extended this research with a number of experiments to study the level of attention loss in intensive dual-task procedures. The results of one experiment found that participants suffered from 'inattention blindness', in that their recognition memory of billboards in a simulated driving environment was impaired while conversing on a mobile phone.

Strayer and Johnston et al. further extended this experiment by tracking the user's eye movement while driving in the simulated driving environment. Their results indicated that even though these participants were directly focused their vision on billboards during a simulated driving task the increased cognitive load of the mobile phone conversation impaired their recognition memory for this information. These results complement those of Sodhi et al. (2002), and Trbovich and Harbluk (2003), who also monitored driver eye movements under conditions of varying cognative demand. Sodhi et al., and Trbovich and Harbluk all monitored driver eye movement when presented with tasks of differing cognitive load.

Sodhi found that during the use of a mobile phone while driving in a real-world environment, the driver's eyes would wander around the centre of the windscreen and glances to the odometer and mirrors were less frequent in comparison to normal driving conditions. Trbovich and Harbluk tested the cognitive distraction of mobile phone use on traffic light awareness, with their results suggesting that while conversing on the phone the number of glances to traffic lights greatly decreased and were in some instances non-existent.

2.4.2. Increased Reaction Times

Alm and Nilsson (1995) conducted a simulated driving study consisting of 40 subjects in order to test the effects of mobile phone use on vehicle following distances. It was concluded that the use of a mobile phone in the simulated environment corresponded to a decrease in reaction time. Moreover, it was found that this impairment was more apparent in older drivers. Alm and Nilsson discovered that the subjects did not compensate for their slowed reaction times by

increasing the following distance to the vehicle ahead. It was concluded that this was due to the fact that the subjects were unaware of the impairment.

2.4.3. Benefits of Mobile Phone Use While Driving?

The aformentioned research has focused on the effects using a mobile phone has on the participants driving ability. However a report by Parkes (1991) used simulated driving environments coupled with in-vehicle driving on urban and rural roads, in order to discover if the task of driving can affect the drivers understanding and interpretation of the mobile phone conversation. It was concluded that participants had significantly greater difficulty in remembering and correctly interpreting information from the conversation while driving, compared to when not driving, or conversing with a passenger.

2.5. Human Computer Interaction in Telematics

It has been shown that the effect of mobile phone use on driving ability is a relevant concern worldwide. However, this issue is not restricted to interaction with a mobile phone. More accurately, use of any telematic devices while driving induces cognitive demands on the driver. Mobile phone use in vehicles has received such publicity as it is the most common case of mobile and telematics device use in vehicles worldwide. Because of this, safety and human computer interaction are fields of major research in the telematics industry.

2.5.1. Base Design Requirements for Telematics Products

Wheatley (2000) has outlined a number of basic design considerations for the development of telematics products, which include;

- Interaction with the system as a secondary task;
- This secondary task should not distract from the primary task of driving by increasing cognitive load on the driver;
- The location of input/output devices must complement the limited space of the driving environment;
- Input/output modalities suit the variable noise environment of the vehicle;

Wheatley (2000) has also outlined a number of human computer interaction design principles, which should be followed in order to develop high-level telematics products, these being:

- Intuitiveness: Consumers expect the device to be easy and simple to use with little to no prior training.
- Consistency: The basis of telematics products should be consistent across different manufactures. Consistency can be helped with the enforcement of industry standards and the implementation of customisable user preferences.
- Interaction Modality: The input and output modalities of the device must suit the environment, and should take into consideration both the useability requirements of the system in the restricted environment of the vehicle, and environmental changes such as variable noise pollution levels.

2.6. Advanced Human Computer Interaction

The human computer interaction requirements outlined by Wheatley (2000) are an integral part of the base design requirements for telematics applications. However with the introduction of advanced user-centric telematics systems in vehicles which can include internet browsing, email and in-vehicle cinema systems, the driver are more likely than ever to be bombarded with information which primarily distracts form the primary task of driving.

2.6.1. Workload Managers

In attempt to combat such distraction Green (2004) has proposed the use of a driving workload manager. A driving workload manager is primarily a context awareness system for the vehicle paradigm. The basic requirement of a driving workload manger is a system that regulates the flow of information to a driver based on the current driving conditions. Workload managers have already been commercially implemented within the automotive industry. The 2003 Saab model 9-3 and 9-5 vehicles contain a ‘dialog manager’ which monitors vehicle speed and windscreen

wiper movement among other things in an attempt to determine when it is safe to present vehicle service reminders to the driver.

Green (2004) states that the enhancement of warning system effectiveness could be an even greater safety benefit than the control of telematics. This enhancement requires minimising information redundancies such as alerting a driver that they have strayed out of a lane, as in most situations the driver will be aware of their offence and the warning will only increase driver distraction. To combat this, the workload manager should attempt to sense driver inattention to the road and only present warnings in situations where the driver is truly unaware.

2.6.2. Peripheral Displays

In an attempt to provide information while not distracting from the primary task has lead to significant research in the field of the peripheral displays.

Peripheral displays are output devices which abstract and present information in a manner which does not interfere with the primary task. Matthews et al. (2004) present a toolkit for managing user attention in peripheral displays. The key features of peripheral displays are data abstraction and notification level, which is associated with the level of available user attention. Data abstraction transforms input data into a form that is 'semantically compatible' with the questions the user is expected to ask of the data (for example). Next, a notification level is selected to alert the user to a change in information. This notification level is associated with the user's currently available attention level. Matthews et al. (2004) conclude that the key for peripheral displays is the way they impact on user attention, which is especially important during mission-critical tasks such as driving a motor vehicle.

2.6.3. Signalling Remote Callers

Strayer and Johnston et al. (2003) found that conversing with a passenger did not increase the cognitive load on a driver to the point associated with inattention blindness, and therefore driving performance was unaffected by conversing with a passenger. It was concluded that in this instance both driver and passenger were aware of the current driving condition and can modulate their conversation accordingly.

Manalavan et al. (2002) extended this idea by creating a platform to signal the remote caller during times of increased driving load. This system consisted of a context-aware mobile phone which signalled the remote caller in times of increased load. It was concluded that the singling of remote callers in times of increased driving load induces the caller to talk less, which in turn lessens the cognitive load on the driver. It was found that when a caller's conversation was reduced there was a marked increase in the driver's performance of the primary task. In simulated driving environments the driver's performance improved to the same level as driving with no phone call.

2.6.4. 'Sensitive' Devices

Hinckley and Horvitz (2001) developed an advanced human computer interaction paradigm by incorporating enhanced notification modalities into mobile devices in order to develop a 'sensitive' mobile device. This study presents a mobile device that is capable of interpreting input from the user by employing sensors that detect when the device is being held by the user, and can detect the location of the device in relation to nearby objects. Hinckley and Horvitz (2001) used these sensors to convey a sense of device sensitivity to the user. For example, when the device would ring to alert the user to an incoming call the sensors would detect when the user touches the device and automatically lower the ring volume.

2.6.5. Gesture In-Vehicle Interface

Alpern and Minardo (2003) explored the use of a gesture interface for in-car control of a secondary task. Their developed system projected an image on to a simulated heads-up-display, which users made gestures with their hand to navigate the interface. It was concluded that for an in-car gesture interface to minimise the effect of distraction of the primary driving task, quick glances of the user's attention must be accommodated. The key design issues were the visibility of options and the ease of navigation.

2.7. In-Vehicle Technologies

Although there is not a clear consensus on which device(s) will emerge as the telematics market leader(s), there has been extensive research to predict, which technologies consumers will wish to have in their vehicle (Figure 2.3).

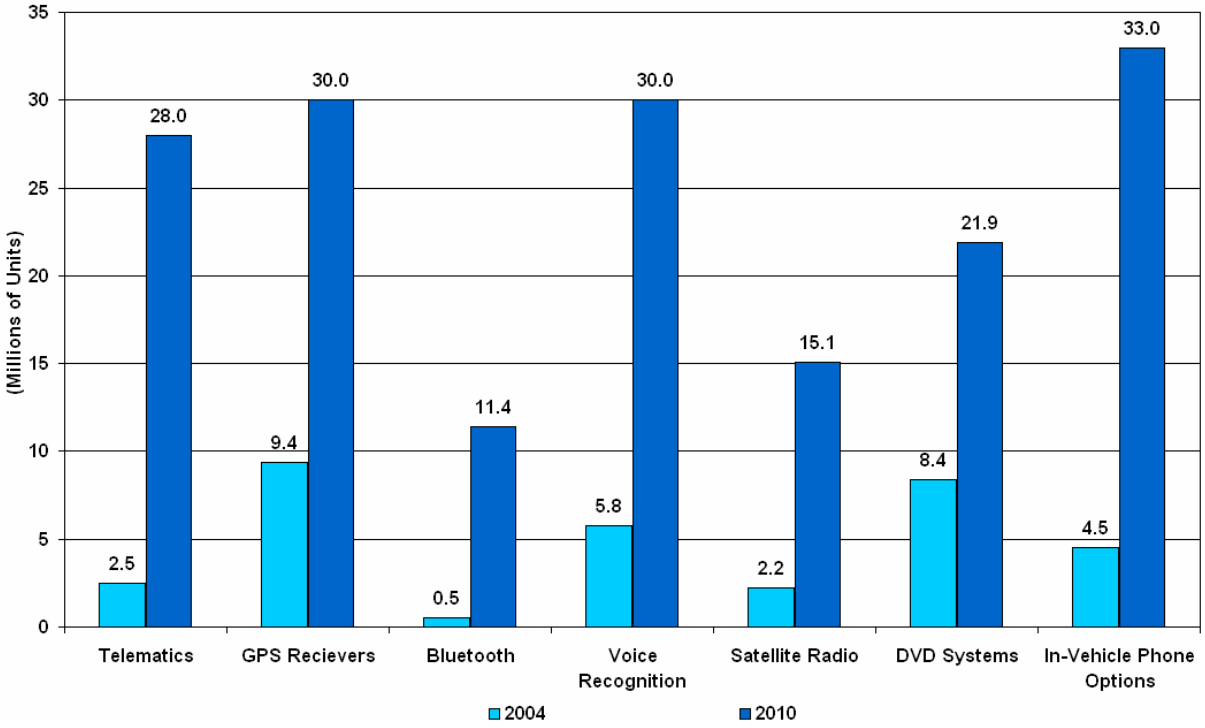


Figure 2.3. Technology Comparison - Global Sales Scenario[4]

Figure 2.3 above shows a market research projection, which estimates that by the year 2010, in-vehicle phone options will be the leading technology in the automotive telematics industry. It is likely that these devices will be coupled with voice recognition and GPS receivers in order to provide a high level interaction modality, and location-based services to the vehicle.

2.8. Mobile Device Integration

Green's (2004) notion of a driving workload manager requires a situation where all devices in a vehicle are interconnected in order for the information and services they provide to be restricted according to the current driving conditions. Moreover, with a

[4] Auto Industry - http://www.autoindustry.co.uk/

strong legal focus worldwide on restricting hand-held mobile phone use while in control of a vehicle, a method for mobile device integration is a key requirement for modern automotive telematics.

2.8.1. What is holding the telematics industry back?

The telematics industry is the amalgamation of multiple industries, all of which are looking to gain from the services they provide. As such, the business case for the deployment of telematics systems is as complex as the systems themselves. The major frontier for the telematics industry is to provide services for converged devices, which can be upgraded and customised as required. This said it is difficult for the automotive industry to corner the market for converged devices. This is primarily due to the development lifecycle of the automotive industry, which is typically 4 to 6 years, where products are developed for 10+ years of use. This is contrasted by the consumer electronics industry where products are designed in 6 to 12 months and are in use for a period of 2 to 3 years (Fuchs and Spaur 2004).

2.8.2. Plug and Play Personal Telematics

Fuchs and Spaur (2004) have proposed a solution to this problem, where automotive manufactures provide services for plug and play personal telematics devices to be integrated into the vehicle. This is achieved by automotive manufactures providing an in-vehicle Telematics Control Unit (TCU: Figure 2.4). The TCU provides access to vehicle resources, including vehicle electronics buses, and access to human machine interfaces, such as in-vehicle displays, audio controls and buttons. Consumer level personal telematics devices then connect to the telematics control unit and can be upgraded as required.

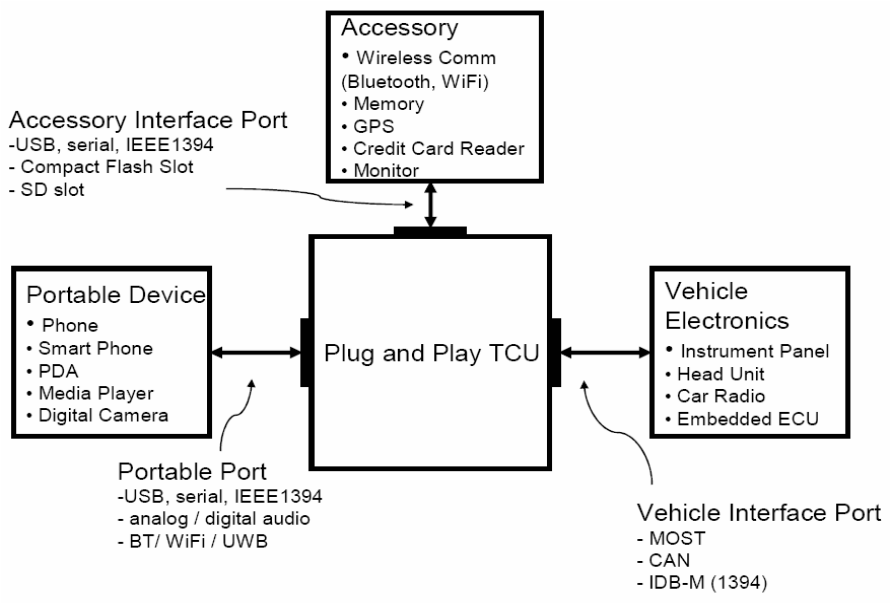


Figure 2.4. Plug and Play Telematics (Fuchs and Spaur 2004)

Figure 2.4 outlines the design of a plug and play telematics control unit and shows how devices are connected to it. Personal telematics devices such as mobile phones and PDAs are connected to the portable port. Shared devices are connected to the accessory interface port, and the vehicle electronics connect to the TCU via the vehicle interface port.

Security is also a major concern in plug and play telematics. In this case the TCU serves as a secure gateway between mission-critical functions on the vehicle interface port and the secondary functions of the consumer devices connected via the portable, and accessory interface port.

This plug and play TCU is integrated with wireless communications mediums including Bluetooth, WiFi and ultra wide band (UWB), this allows a wide range of devices to be connected to it, without having to provide specialised interfaces for connection of proprietary devices.

There are however, a number of issues in allowing personal plug and play telematics devices to connect to the vehicle. A major requirement for a platform such as this is a common software framework between the personal telematics device and the TCU.

Open Operating Environment

It is unreasonable to expect all in-vehicle devices and personal telematics devices will have the same software operating environment. As such, a successful plug and play personal telematics system connectivity framework must allow for connection from a wide variety of standardised open environments such as the Open Services Gateway Initiative (OSGi), and the Automotive Multimedia Interface Collaboration (AMI-C).

Microsoft T-Box

This said the open source versus proprietary closed system debate has found its way to the telematics platform. Microsoft has developed the T-Box, a plug and play telematics platform developed for the Microsoft Windows Automotive operating system. The T-Box is designed to communicate with personal telematics devices running the Windows Mobile for Automotive operating system. Windows Mobile for Automotive is a low end standard system that provides a gateway for entertainment, mobile phone and other devices brought into the vehicle to connect to the T-Box, while also offering a standardised platform for OEMs to build features that are specific to assessing the vehicle bus. The T-Box allows voice control of a mobile phone, or other personal telematics device running Windows Mobile for Automotive (Microsoft 2005).

2.9. Advanced Telematics Systems

As research has increased in the area of telematics and mobile device integration there have been multiple implementations (some commercial, some research based) which employ advanced human computer interaction techniques and input/output modalities to attempt to provide a suitable interaction medium for drivers while not interfering with the primary task of driving.

2.9.1. SmartKom

Human-computer interaction in a restricted environment such as the vehicle is a difficult issue to overcome. The German based SmartKom project provides a system that utilizes advanced multimodal human computer interaction (Malaka, Haeussler and Aras 2004), (Reithinger et al. 2003), (Jöst et al. 2005). The SmartKom system realises full symmetric multimodal interaction, in that all input mediums can also be

used for output. The system is designed to be device independent by allowing users to connect their devices to the system to create a 'personal IT infrastructure'. The system designers were aware that for a system to be 'intelligent' it must be able to interpret the context of the user's current situation. Because of this the SmartKom system is segmented into three 'environments' home: public: and mobile, with the mobile environment incorporating sub-environments of car and pedestrian. The purpose of these 'environments' is to provide the most appropriate input/output modalities for the current user context. For example in the car environment the dominant modality is speech, so as to limit the level of physical interaction of the driver.

This said the SmartKom system also allows for flexible device management, this permits the user to choose the device with which they interact. For example in the car environment the user may select either the screen of the navigation system for visual system output, or another device such as a personal digital assistant (PDA).

2.9.2. Linguatronic

As seen in Figure 2.2 mobile device integration is the leading telematics technology in Europe. The Linguatronic system was designed so drivers could interact conduct complex interactions with in-vehicle systems and mobile devices.

The Linguatronic system is a voice operated Command & Control system that was first deployed in 1996 in the S-Class series of Mercedes-Benz cars in Germany (Bühler et al. 2003), (Heisterkamp 2000). The Linguatronic system allows for complete hands-free operation of the vehicle's mobile phone, including number dialling, number storage, user defined telephone directory, name dialling and directory editing. The Linguatronic I Command & Control system contained a vocabulary of 30 speaker independent words, which included digits and control words. The second generation of the Linguatronic extended the vocabulary to approximately 300 words, which enables the voice control of the vehicles electronic devices such as the radio and air-conditioning. This is made possible by connecting the system to the vehicles optical fibre data bus. This data bus is the central information channel that connects all devices in the vehicle. As the speech modality can be used to control the vehicles electronic devices, the Linguatronic system

increases the useability of the vehicle computer interface. The Linguatronic system features a push-to-activate (PTA) button that signals that a command is to be entered.

SpeechDat-Car

One issue with voice based in-vehicle systems for mobile device integration, particularly in Europe, is the need for the system to support multiple languages for input and output. SpeechDat-Car is a project focused on the development of a set of speech databases to support the training and testing of multimodal speech recognition applications in the vehicle environment (Heuvel et al. 1999). The SpeechDat-Car project commenced in 1999 and has lead to the development of speech databases in nine languages, developed by recording speakers in different typical vehicle noise situations. It is the development of projects such as this which lead to the increased development of speech based in-car systems.

In Silico Vox

The 'In Silico Vox' project is a joint venture between the Carnegie Mellon University and the University of Berkeley. The purpose of this project is to implement a speech recognition hardware device. The development of speech recognition in hardware requires less computational processing and is therefore much more efficient than their software-based counterparts (economist.com 2005). Additionally, with the adoption of speech recognition hardware chips, advanced speech modalities can be incorporated into devices with less computational ability such as mobile phones. This could lead to a new revolution of voice-based human-computer interaction.

2.10. In-Vehicle Networking

The plug and play telematics control unit defined by Fuchs and Spaur (2004) in section 2.8.2 enabled consumer mobile devices to be connected to the vehicle using a wide range of media. Theses included both wireless mediums such as Bluetooth, WiFi and Ultra wideband and wired mediums including USB, serial and IEEE 1394 (FireWire). The telematics control unit described by Fuchs and Spaur (2004) envisioned that wired mediums would be used to connect devices with different physical interfaces. A similar issue is relevant for connection of wireless devices to

the control unit. However, why do we need multiple wireless mediums in the vehicle? The simple answer is that these technologies provide different services.

2.10.1. Bluetooth

Figure 2.3 shows that the availability of Bluetooth in the vehicle environment is anticipated to grow steadily until the year 2010. Simply stated Bluetooth is a short-range communications protocol the purpose of which is to act as a cable replacement technology by connecting low powered devices without the need for proprietary cables. The Bluetooth standard outlines a signal range of up to 100 meters, which depends on the category of device (Hopkins and Antony 2003), with low powered devices such as PDAs and mobile phones, for example, usually operate at a range up to 10 meters. Bluetooth operates in the 2.4 GHz ISM frequency band. and transmits at a raw data rate of 1 MBps. Transmission at this relatively low speed, at a distance of up to 10 meters means that Bluetooth has a low power requirement, which is suited as a communications medium between battery powered mobile devices.

2.10.2. WiFi

WiFi is the collective umbrella under which 3 major wireless local area networking (WLAN) technologies fall. These are IEEE 802.11a, 802.11b and 802.11g. These technologies can transmit at data rates from 11 MBps (802.11b) to 54 MBps (802.11a) (WiFi Alliance). These technologies are designed to connect relatively high powered devices, and therefore have a higher power requirement than that of Bluetooth. 802.11b operates in the 5 GHz ISM band. However 802.11a operates in the 2.4 GHz and therefore has the potential to interfere with Bluetooth.

2.10.3. Bluetooth and WiFi

Chiasserini and Rao (2000) explored coexistence mechanisms as a solution to the interference in concurrently running both IEEE 802.11b and Bluetooth in the 2.4 GHz ISM frequency band. They explored both collaborative and non-collaborative methods. Collaborative methods require both Bluetooth and 802.11b transmitters reside in the same terminal. Collaborative methods use scheduling methodologies to restrict the transmission of either technology. This can be achieved by using a method such as Time Division Multiple Access (TDMA), where both technologies are scheduled to transmit at different time intervals and therefore never overlap.

However, non-collaborative methods contain no way of communicating this scheduling information between systems running different technologies. Therefore the coexistence mechanism must be able to operate independently. An example of a non-collaborative coexistence mechanism is the Adaptive Frequency Hopping technique. In this method the frequency channels are partitioned and classified as either ‘good’ or ‘bad’. If the device selects a channel which is ‘bad’ it is replaced with a ‘good’ channel from the pool.

The aforementioned research has outlined the current state of the automotive telematics industry. However, in order to provide security services in the telematics environment an introduction into the field of security is required.

2.11. Authentication

The act of authentication in the vehicle environment may be the defining factor in the acceptance of telematics devices. If users can authenticate themselves transparently, or with minimal physical interaction with the system they are more likely to use the system, which in the vehicle paradigm is designed to provide safer interaction with the device. Moreover, when a telematics system can successfully authenticate users, it can then offer a greater range of user specific services from user define preferences to user sensitive data such as email.

2.11.1. Traditional Authentication

Schneier (2004) describes three ways to authenticate a user. By something the user knows, for example a password. By something the person has in their possession, for example an authentication token, an identification card, or even the SIM card in a mobile phone. Finally you can authenticate a user by using something specific about the person themselves.

Authenticating a user by using something a person knows is the most widely known method of user authentication to an operating system. This method binds a unique username representing a user’s identity, with a password, which acts as a shared secret between the user and the system. However, in a networked world users cannot afford to transmit this shared secret across an unsecured network. As such, the

requirement for systems to employ methods to hide these shared secrets has arisen. Additionally the operating system itself requires a list of the passwords for all users that are authorised to access the system. This list could also be vulnerable to attack. Because of this, each host in the network required a facility to mask a user's password, which could then be compared to a directory of passwords for all users masked in the same way.

2.12. Cryptography

Cryptography is the process of transforming a user's password into a form where it is unrecognisable as its original form. Menezes et al. (1997) define the field of cryptography as *"the study of mathematical techniques related to aspects of information security such as confidentiality, data integrity, entity authentication, and data origin authentication."* Cryptography is traditionally a mathematical field, where cryptographers use well-proven mathematical functions to develop methods to convert plaintext information into ciphertext where this ciphertext cannot be reengineered to the original plaintext without secret knowledge, where this 'secret knowledge' is known as a key. In a modern world the field of cryptography encompasses a broad range of disciplines ranging from mathematics, to computer security and even civil and criminal law among others.

2.12.1. Goals of Cryptography

Menezes et al. (1997) nominate four objectives which form a framework for information security. These include privacy or confidentiality, data integrity, authentication and non-repudiation.

Confidentiality

The objective of confidentiality enforces a requirement where the content of information must be kept from all those except those authorised to have it. From a cryptographic perspective there are many approaches to facilitate the conservation of confidentiality from physical protection to mathematical algorithms which transform data so it is unintelligible.

Data Integrity

The objective of the data integrity requirement is to guarantee that data has not been modified by an unauthorised entity. To ensure data integrity the ability to detect any modification to data by unauthorised parties is required. Data integrity must be able to check for unauthorised insertion, deletion and substitution of data.

Authentication

The objective of authentication is to prove the identity of the sender of the information and the information itself. Information which is transmitted over a channel should be authenticated as to its origin and include other unique identifiers including date and time stamps, random numbers, or the like. From a cryptographic perspective authentication is concerned with both entity authentication (proving the identity of the sender or originator of the information) and data origin authentication (which provides data integrity).

Non-Repudiation

The objective of non-repudiation is to prevent an entity from denying previous authorised communication. In this instance the services of an independent trusted third-party are often employed to resolve disputes.

As mentioned above the field of cryptography contains of many different algorithms and techniques all of which are designed with differing goals in mind and are therefore used in different situations.

2.12.2. Encryption

Schneier (2003) states that encryption was the original goal of cryptography. Encryption allows a message to be sent between two entities on an unsecured channel who share a secret. The requirement of the shared secret means the contents of the message is secured from a third-party who eavesdrops on the channel, who does not share the secret. This process involves converting a plaintext message *(m)* by use of an encryption function described as $c := E(K_e, m)$ which produces a ciphertext *(c)* result. Both entities share the secret K_e. Therefore when one trusted party encrypts a message and creates ciphertext *(c)* then the other entity can perform

use the decryption function described by $m := D(K_e, c)$ to return the original plaintext message *m* (Schneier and Ferguson 2003).

The method for encryption described above is an integral step to prove the identity of an entity. This is why encryption methods are strongly associated with authentication.

2.12.3. Message Authentication Codes

However, the problem of authentication is only partly solved by encryption, in that anyone listening on the channel could still intercept the message sent between the two authorised entities and modify this message in some way.

To resolve this limitation, Message Authentication Codes (MACs) come to the fore. A MAC is a function similar in design to the encryption function described in section 2.12.2. However, a MAC function is different to an encryption function in that it is designed to protect the integrity of the message. In this instance, if two entities wish to transmit a message securely across an unsecured channel the sender passes a plaintext message *(m)* through a MAC function *(a)* as described in $a := h(K_a, m)$, where *h* is the MAC function and K_a is the shared secret between the two entities (Schneier and Ferguson 2003). In this case the sender transmits both the plaintext message *(m)* and the MAC output *(a)* to the receiving entity. The message recipient then recomputes the MAC *(a)* and if the two versions of the MAC *(a)* are identical then the message has not been modified during transmission. Moreover, if an entity eavesdropping on the channel was to intercept the message and replace, or modify the message *(m)* in any way, the value the receiving entity computes for the MAC function *(a)* will be different to the one in the message received. Therefore , if the two MACs *(a)* are identical, the message receiver can be assured that the message has arrived in the form in which it was sent.

Data which is protected by the use encryption and message authentication codes can still be susceptible to attack, as in this case there is nothing stopping an unauthorised party from either deleting or replaying a legitimate message. There are a number of methodologies to protect against these types of message stream modification attacks. These include the use of a sequence number or a timestamp. In this case each

message can contain a sequence number or a timestamp. When the receiving entity accepts a message the sequence number or timestamp is checked either, with the requirement either, that it is strictly greater than that of any message received so far, or with a timestamp the time or data origin can be determined for validation of the integrity of the information. Additionally a timeliness requirement can be added to messages, so that messages with a timestamp value outside of a predetermined range are automatically discarded. In both cases there are methods in place to limit susceptibility to message replay attacks.

2.13. Key Distribution Problem

Sections 2.12.2 and 2.12.3 have both made reference to a secret shared between the two conversing entities, in the form of a key. However, as yet there has been no method discussed for the creation or distribution of these keys.

Schneier (2003) states the issue of managing and distributing keys is one of the major issues in the field of cryptography. In most cases, the physical distribution of keys is impractical as they are largely designed to be used by systems in a networked environment. Due to this, one to one physical contact between two entities wishing to share a key is usually impractical.

2.13.1. Symmetric-Key Encryption

Symmetric or secret key encryption it the simplest form of encryption. The shared key as used in sections 2.12.2 and 2.12.3 are both symmetric keys in that the same key is used for both encryption and decryption. Because of this the encryption and decryption algorithms used in symmetric key encryption are extremely close in design. Pfleeger (2003) states that as long as the key remains a secret, symmetric key systems can provide authentication, and ensure a base level of integrity as the original message will not decrypt properly if it has been modified in transmission.

Symmetric key encryption is acceptable if there is a secure method to generate and distribute keys in such a way that only authorised users gain knowledge of this key. However, in a networked environment where secure communication is required between entities that are not necessarily known, how do entities establish a link between themselves where a symmetric key can be securely shared?

2.13.2. Public-Key Encryption

A solution to this problem is public or asymmetric key encryption. In this model each entity generates a pair of keys (S_1, P_1) using a defined algorithm, where the keys created consist of a secret key (S_1) and a public key (P_1) (Schneier and Ferguson 2003). The differing factor in public-key encryption as apposed to the symmetric-key encryption methodology described in section 2.13.1, is that in public-key encryption each entity publishes their public key. Therefore, if an entity wishes to communicate with another entity in a secure fashion, they must first look up the other entities public key (P_1) and use this key to encrypt a plaintext message (m) which creates a ciphertext (c), which is then sent to the owner of public key (P_1). This entity can now use their secret key (S_1) to decrypt the ciphertext (c) which will return the plaintext message (m) if the message has not been tampered with in transmission. Therefore the algorithms for key generation, encryption and decryption must ensure that decryption returns the original plaintext. Therefore Schneier (2003) defines public-key encryption as $D(S_1, E(P_1,m)) = m$, for all m.

2.13.3. Digital Signatures

Schneier (2003) states that digital signatures *"are the public-key equivalent of message authentication codes (MACs)."* This is because like MACs, digital signatures are concerned with ensuring message integrity. In this instance entities are still required to have generated a pair of keys consisting of a public key (P_1) and a secret key (S_1), where the public key for each entity must be publicly available. In the case of digital signatures, when an entity wishes to bind their identity to a message they sign a plaintext message (m) with their secret key (S_1) as defined in $s:=\sigma(S_1,m)$. The next step is to send the signed message (s), and the original message (m) to the receiver, who can ascertain who the sender was by using a verification algorithm along with the senders public key as defined by $\upsilon := (P_1, m, s)$ in order to validate the signature. The process of signing a message and then verifying the origin of the message is handled in the same way as for message authentication codes (MACs). However, in this case a receiving entity can verify the origin of a message by using the sender's public key, where the sender must use their secret key to sign the message. Therefore an entity can verify the identity of the sender by

using a piece of public knowledge *(P_1)*, to verify the signed message *(s)*. Moreover, anyone else on the channel can do this not just the receiving entity.

The goal of a digital signature is to bind identity to a message by signing it with a piece of secret knowledge *(S_1)*. As this knowledge is secret, then digital signatures can be used to enforce non-repudiation, where the entity who signed the message cannot deny signing the message as only they have access to their secret key *(S_1)*.

2.13.4. Evaluation of Encryption Methodologies

There are positives and negatives for both symmetric-key and public-key encryption schemes. On one hand key distribution in public-key encryption methods is considerably simpler than that of symmetric-key encryption. However, public-key encryption endures a greater computational load than that of symmetric-key encryption. Moreover, public-key encryption is drastically less efficient than symmetric-key encryption, due to the inefficiencies and the requirement for public-key encryption schemes to utilise three distinct algorithms for key generation, encryption and decryption.

2.13.5. Evaluation of Message Integrity Methodologies

Much the same as there are positives and negatives for both, symmetric key and public key encryption methodologies, the same holds true for the methodologies that ensure message integrity. However, as stated in Section 2.13.4 the main issue with the encryption schemes is efficiency. In this case message authentication codes (MACs) which employ symmetric key techniques to hide the contents of a message are inherently more efficient than the public key techniques used in digital signatures. This is primarily due to the fact that digital signatures require the use of multiple algorithms for key generation, message signing and message verification.

2.14. Cryptographic Functions and Algorithms

Sections 2.12 and 2.13 provided an introduction to the field of cryptography and the goals and issues solved by differing cryptographic methodologies. The following section will provide an overview of the major cryptographic algorithms.

2.14.1. Block Ciphers

Block ciphers are an integral building block of modern cryptography. Schneier (2003, p. 43) defines block ciphers simply as an encryption function for fix-sized blocks. Traditionally block cipher algorithms utilise symmetric key methodologies to encrypt a message of *n* length with a shared key to produce a ciphertext. Decryption is achieved in the opposite fashion, in that the ciphertext is decrypted with the same shared key as used for encryption. This function will produce the original plaintext message, as long as the ciphertext was not tampered with in transit. The most widely known block cipher algorithm is the Digital Encryption Standard (DES). However, recent advances in cryptography and computer processing speed have rendered DES largely useless due to its restricted key size and small block size of 64 bits. Because of this small block size and mediocre key length the security offered by DES is not sufficient for modern security applications. A possible solution to strengthening DES was to expand the operation of DES into a scheme known as 3DES which was simply 3 DES encryptions running in sequence. However, the enhanced length of 3DES does not heighten its strength to an acceptable level for modern security applications.

The problem that was left unsolved by the attempted strengthening of DES was that the underlying algorithm was left largely unchanged. Therefore, if a weakness was discovered in the underlying algorithm, any subsequent methods built on this algorithm were likely to be exploited in the same way.

A solution to this was to reengineer the underlying block cipher algorithm. This was the achievement of the Advanced Encryption Standard (AES). This method uses a different approach than DES, where each step of the algorithm includes a number of operations which can be performed in parallel, which greatly enhances the speed of the algorithm

2.14.2. Hash Functions

Schneier (2003, p. 83) states that hash function are among the most versatile of the cryptographic primitives. This is due to the ability of hash functions to be used for encryption, authentication and for message integrity.

Hash functions differ to block ciphers in that they produce a fixed-size *h(m)* output from an arbitrarily long input *(m)*. This fixed-sized output, coupled with efficient one-way operation makes hash functions the most versatile of the cryptographic primitives. The basic requirement for a successful hash function is that the function must be one-way, in that the original plaintext *(m)* can not be found given the hash output *h(m)*. Another requirement for a successful hash function is the level of collision resistance, in that the hashed output *h(m)* must be unique for every different value of *(m)*. However this requirement has can never hold true as hash functions return a fixed length output there are an inevitable number of collision as an infinite number of possible inputs *(m)* are constrained to return a finite number of outputs *h(m)*.

The two most popular hash functions are the message-digest algorithm MD5 and the secure hash algorithm SHA-1. MD5 is a hash function which returns a hash of 128 bits or 16 bytes, where SHA-1 returns a hash of 160 bits or 20 bytes. In each case the plaintext message is segmented into fixed size blocks for processing.

2.14.3. Hash Message Authentication Code (HMAC)

Stallings (2003b) provides three motivations for the combination of hash functions and message authentication codes. These consist of the following:

1. Cryptographic hash functions such as MD5 and SHA-1 execute faster in software than block ciphers such as DES.
2. Documented code is widely available for these peer reviewed hash functions.
3. The export restrictions in US that cover block ciphers algorithms when used even to compute MACs do not apply to hash functions.

This said, hash functions such as MD5 and SHA-1 were not designed to be used as a message authentication codes (MACs) as they do not rely on the use of a secret key. The most widely accepted solution to this problem is HMAC as defined in RFC2104 (Krawczyk, Bellare and Canetti 1997). HMAC has been chosen as the mandatory-to-implement MAC for IP security, and is also used in other protocols including SSL and SNMP. HMAC employs trusted hashing algorithms which perform well in

software such as MD5 and SHA-1. Moreover, HMAC defines a way that the underlying hash function (MD5 or SHA-1) can be used unmodified, and provides a facility to change the underlying hash function with no manipulation of the HMAC algorithm required. HMAC process data in 64 bit blocks and returns a value equal to the byte length returned by the underlying hash function. Therefore HMAC would return a value of byte length value of 16 (16 bytes) for MD5 and 20 (20 bytes) for SHA-1. The keys used in HMAC are generated using a secure method and should not be smaller than the byte length returned by the underlying hash function, and should not be larger than the block length (64 bits). Keys that are larger than the block length (64 bits) are hashed using the underlying hash function. In this case the resulting value is used as the key.

As HMAC utilises well tested hash functions, the level of security in HMAC is dependent on the level of security offered by the underlying hash function. As HMAC employs the services of both hash functions and message authentication codes (MACs), HMAC can be used for encryption, authentication and for message integrity.

2.15. Key Generation and Exchange

The aforementioned sections have focused on the use of a shared key. However, as yet there has been no method outlined to provide a secure method for this key to be exchanged between two entities communicating across a public channel.

2.15.1. Diffie-Hellman Key Exchange

The Diffie-Hellman key exchange algorithm was the first method for secure key exchange between two entities communicating across a public channel. This method was developed by the visionary cryptographers Whitfield Diffie and Martin Hellman (Schneier and Ferguson 2003). This method utilises the services of public-key cryptography in that like other public-key architectures, it requires each entity to generate a public and secret key pair and make their public-key freely available. However, unlike traditional public-key methods these keys are not used for encryption and decryption, they are used to create a single shared secret between entities.

The Diffie-Hellman protocol is the combination of one entities secret key with the public key of the communicating entity in order to create a shared secret. This process begins with the two communicating entities agreeing on a large prime number *n* and *g* such that *g* is primitive mod *n*. These numbers do not need to be secret, and therefore can be agreed to over an unsecured channel (Schneier 1996). Next, both entities choose a random large integer and send this to the communicating entity. The calculation on these shared numbers provides a way to combine one entities shared key with the public key of a communicating entity to create in essence a shared symmetric key.

2.16. Cryptography on Mobile Devices

Implementing cryptographic algorithms on constrained mobile devices is a grate challenge. This is due to mobile devices having dramatically less processing power, memory, and storage than even the most basic desktop machine. This is why the protocols which have been developed to secure personal computers and networks can not be directly applied to the mobile platform. This section will provide an insight into the existing research into the notion of mobile device cryptography, and will focus on the methodologies for key generation and exchange.

Jakobsson and Pointcheval (2002) define a method for mutual authentication for low-power mobile devices. This research details a study of protocols for mutual authentication and key exchange aimed for mobile devices based on the public-key exchange protocol defined by Diffie-Hellman as outlined in section 2.15.1. This paper presents a model where computation is kept to a minimum on constrained devices. Moreover this paper has concluded that the storage requirement for the developed protocol is only 70 bytes per process.

Another piece of research in this area was conducted by Chou (2003). This research also experimented with public key cryptography and key exchange based on the Diffie-Hellman key exchange method. However, this research utilised elliptic curve algorithms to provide a public key architecture which executes faster in software than traditional public key systems. This is due to elliptical curve cryptography providing equal or greater security than traditional standards using a smaller key

size, which in turn reduces the processing overhead (Stallings 2003b). Therefore elliptic curve operations execute quickly in software.

2.17. Biometrics

Authenticating a user by using something specific to the user themselves forms the basis of the field of biometrics. Schneier (2004) defines biometrics as *"something the person has that's a physical part of their body"*. Biometric authentication is a familiar concept, as, for example, we all recognise friends and family by their distinguishing physical attributes or through the sound of their voice. Biometrics has the distinct advantage over other traditional authentication methods, as a biometric is not susceptible to 'forgetting' by a user, in contrast to the use of passwords. Biometrics disciplines include a multitude of fields including:

- Face Recognition
- Finger Print Recognition
- Hand Geometry
- DNA
- Iris Scanning
- Signature Geometry
- Retina Scanning

The disciplines mentioned above contain both obtrusive and unobtrusive methods of authentication. Obtrusive methods require the user to take the effort to make their biometric available for use by the system. Conversely, unobtrusive biometrics can operate in the background and are therefore transparent to the user. Therefore, face recognition has the potential to be one of the most unobtrusive and transparent methods for biometric authentication.

2.17.1. Face Recognition

Tolba et al. (2005, p. 1) defines face recognition as *"a biometric approach that employs automated methods to verify or recognise the identity of a living person based on their physiological characteristics."* Face recognition has the advantage over other biometrics as it is a passive, unobtrusive and transparent way to verify identity in a natural manner. Face recognition systems consist of three major steps.

Firstly a sensor takes an observation of the subject. Next the observation is normalised into a signature by a given algorithm. Finally, this normalised signature is compared against a database of stored signatures and is given a similarity score, although this similarity score is implementation dependent.

2.17.2. Eigenfaces

There are many approaches to face recognition, all which employ different techniques. One of the most popular and widely used approaches is the eigenfaces technique. Eigenfaces uses principal component analysis to represent pictures of faces. This approach approximates facial geometry by calculating a number of weights for each face which are obtained by projecting the face image onto a standard face picture known as the eigenpicture. The eigenfaces method first normalises the images provided for training. This normalisation process locates facial features including the eyes, nose and mouth. Next the normalised images are processed into eigenvectors which are simply a two-dimensional array of weights calculated by mapping the normalised image onto the eigenpicture.

2.17.3. Biometrics for Transparent Identification

Ailisto et al. (2004) proposed a system where multimodal light biometrics were used to provide unobtrusive user identification. This system looked at using a subject's height, weight and body fat percentage as an identification mechanism. It was concluded that using a single light biometrics was not sufficient to provide an accurate enough system. Therefore, in a test group of 62 subjects an 11 percent error rate was achieved when height was the only modality measured. However this rate was decreased to just 2.4 percent when coupled with height data. The authors' state that this method of unobtrusive user identification would only be applicable in a non-secure environment, as data used for identification in this process could be easily forged.

2.18. Network Security

In order for a telematics system to be secure it must be able to successfully authenticate both users and network devices. Networks by default are inherently insecure. With the advent of malicious users, modern networks must employ certain security measures in order to guarantee the confidentiality, integrity and availability

of services and user data. The measures used to secure networks are based on circumventing the following threats and attacks.

2.18.1. Confidentiality Threats

Confidentiality threats are concerned with stopping unauthorised access to sensitive data.

Exposure

Exposure is a threat to the confidentiality of data in that any message which is intercepted wheather at its source, destination or at any intermediate node in the network can lead to the exposure of sensitive information.

Traffic Flow Analysis

Sometimes not only is the message itself sensitive, but the fact that the message exists can also be sensitive.

2.18.2. Integrity Threats

Integrity threats are concerned with making sure that every piece of data remain consistent with the manner in which it was sent by the source, and has not been modified in any way.

Falsification of Messages

If traffic arrives at a destination addressed from a trusted source the user will more than likely assume that the communication was sent from the trusted party and that it has not been unaltered. It is this level of implicit trust that attackers can take advantage of. Pfleeger (2003) has outlined the following threats which an attacker could employ to falsify the integrity of information. An attacker may:

- Change some/all of the content of a message;
- Replace a message entirely, including the date, time and sender/receiver information;
- Reuse (replay) an old message;
- Combine pieces of different messages into one;
- Change the apparent source of a message;
- Redirect a message;

- Destroy or delete a message;

2.18.3. Availability Threats

Availability threats are concerned with the availability of data and services upon demand by an authorised user

Denial of Service

If an attacker can block selected or all traffic to a destination or from a source then this can result in a lack of availability.

The aforementioned threats outline the multitude of ways that attackers can gain access to confidential information, modify data and masquerade as a trusted source. For these reasons when administering a network remotely the administrator must be assured that they are issuing commands in a secure impenetrable environment. Therefore the network administrator must employ the services of a network management tool.

2.19. SNMP

Simple Network Management Protocol (SNMP) is the most widely used network management tool for TCP/IP based networks. By definition SNMP is designed to be 'simple' and is therefore easy to implement and requires little computational processing and very few network resources. SNMP is designed to operate at the application level of the TCP/IP suite, and operates over the user datagram protocol (UDP).

2.19.1. SNMP History

SNMP was originally specified in 1988. The original version of this specification, referred to as SNMPv1 contained security deficiencies including a lack of authentication and privacy mechanisms. This version was soon upgraded to SNMPv2, which included a security facility. This security facility utilized an insecure password-based A further revision of this specification, SNMPv3 was incorporated. SNMPv3 is not a replacement for SNMPv1 or SNMPv2, but is rather an extension to the predefined framework. SNMPv3 contains a security capability which is to be used along side SNMPv2 or SNMPv1. The aforementioned security

capability of SNMPv3 provides a facility for network devices to communicate securely, thus providing a rigid platform for distributed network management.

2.19.2. Network Management Systems

Network management systems are comprised of two types of systems, agents and managers. Any device in the network which is to be managed must include an agent module. The agent module collects and maintains information regarding its local environment and provides this information to a manager when needed, as well as responding to manager commands and altering local configurations accordingly. A network management system will also contain one or more management stations. These management stations include a user interface to provide a facility to observe network status, and for issuing commands to agents. Each agent maintains its own management information base (MIB). The MIB contains past and present information regarding the local configuration and traffic information. The management station, in contrast, contains a global MIB which consists of summary data from all agents. For a stand-alone management station a manager process is used to control access to the global MIB at the management station. This manager process achieves network management by using SNMP which is implemented on top of UDP, IP and the utilized network protocol (Stallings 2003a, p. 253).

Stallings (2003a) suggests that SNMP is considered to be 'simple' because its limited to provide only four functions.

- Get: Used by a manager to query an agent's MIB and retrieve an item.
- Set: Used by a manager to set a value in an agents MIB.
- Trap: Used by an agent to send an alert to a manager.
- Inform: Used by a manager to send an alert message to another manager.

2.19.3. SNMP User Security Model (USM)

The User Security Model is the facility which provides authentication and privacy services for SNMP. The USM is in place to protect the confidentiality, integrity and availability of data in transit between SNMP entities. The USM does this by protecting against unauthorised modification of messages in transit. It also protects against masquerade, disclosure, and replay attacks. However, the USM does not

protect against denial of service attacks or traffic analysis (Blumenthal and Wijnen 1999).

2.19.4. SNMP Key Localization

It is a requirement of SNMPv3 that any communication between a management device and an agent device must include a secret authentication key and a secret privacy key, which must be shared between the devices.

To simplify key management for users SNMPv3 utilises an algorithm for using a user's password to generate keys. Because of this there is no need for the network management system to store user's keys, as user keys are generated from that user's password when needed. Also, a single password can be used to generate keys for both authentication and encryption.

A localised key is defined as a secret key shared between a management device and an agent device. Key localization is concerned with creating keys which are unique for all devices. Finally, the localization key must be configured on the agent system in 'some secure fashion' (Stallings 1998).

SNMPv3 has the advantage over other methods of network security as SNMPv3 employs the services of symmetric-key cryptography which can be used to efficiently generate shared secrets for use in a networked environment.

2.20. Conclusion

This chapter has provided an introduction into the telematics industry and the requirements for developing applications for the vehicle domain. Moreover this chapter has provided a broad level of background into the inner workings of different and competing security methodologies.

3

3. Methodology

Chapter 2 provided an overview of the telematics industry and the challenges of designing systems for the vehicle paradigm. However, as was shown and the previous section, these systems have largely focused on advanced human computer interaction techniques and other matters at the price of security.

The goal of this thesis is to develop a two-tier security architecture for the emulation of telematics systems and mobile device integration. In order for mobile devices to be successfully integrated into vehicles, a security architecture is required to make sure that only authorised devices can interact with the system.

The proposed solution is based around two situations. The first being an organisation with shared vehicles, where all staff are provided with a Bluetooth enabled mobile device. User preferences can be stored on the device and uploaded to the vehicle when the device is authenticated. This allows each user to store preferences including seat and steering wheel position, and even radio station presets in their mobile device. The second situation is an emulation of modern telematics systems. In this case telematics systems can be emulated by interconnecting consumer level devices in a scatter-net architecture, as shown in figure 3.1. The proposed solution enables telematics systems to be developed at a lower cost than commercial telematics systems. Moreover systems can be designed to specific user specifications.

Security is paramount in both of these situations as the mobile phone is the device which enables the emulated telematics system to be connected to real-time services, and provide access to mobile voice/data services.

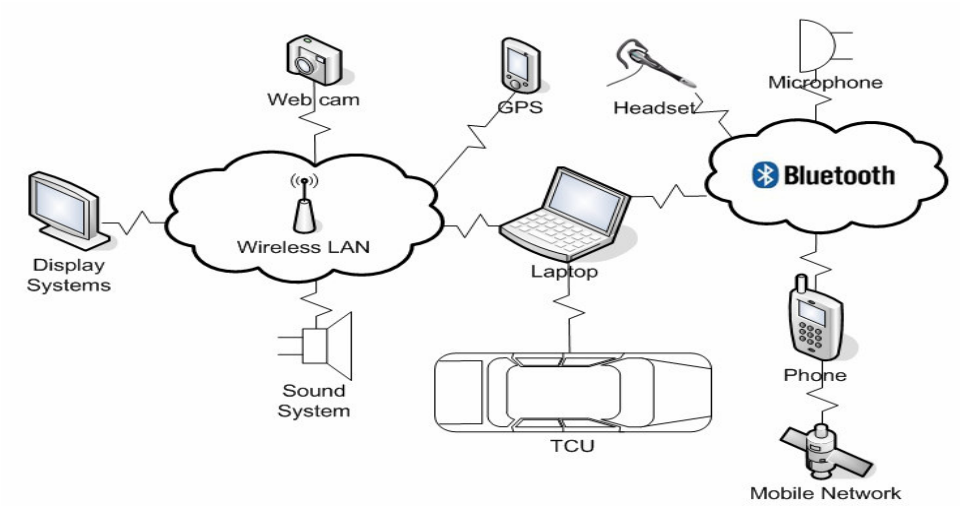


Figure 3.1. Telematics emulation using consumer level devices

Figure 3.1 above illustrates a topological diagram of a typical emulated telematics system using consumer level devices. This emulated system conforms to the traditional design requirements of a telematics system as outline in Section 2.2. In this system the laptop is the main device and is also the device which interfaces with the vehicles onboard peripherals via a telematics control unit as outlined in Section 2.8.2. This topology also contains a global positioning system (GPS) to enable location-based and tracking services for the vehicle. This system also conforms to the human-computer interaction requirement of speech based interaction for the vehicle paradigm as discussed in Section 2.7 by means of the connected microphone for input and utilising the sound system for output. Moreover, this system consists of both Bluetooth and 802.11 wireless LAN mediums for interconnection with other devices in the system. This system can also employ the services of mobile voice/data networks such as GSM/GPRS or CDMA/EV-DO via the mobile phone that acts as a modem.

The development of a two-tier security architecture for mobile device integration requires three main steps. First, a device authentication mechanism must be developed. Second, this mechanism must be developed into a protocol in order for it to function in a networked environment. Third, a method of user authentication or user verification must be developed to define the user's authority to use the authenticated device.

3.1. Device Authentication

In a shared vehicle organisation where user preferences are stored on each user's mobile device, it is paramount that the user authenticates their device before they are allowed to access user specific services in the vehicle. This need is enhanced if personal preferences are stored on the device such as passwords or email authentication details. The need for device authentication in an emulated telematics environment is largely an issue of financial cost. In the emulated telematics topology depicted in Figure 3.1 the mobile phone is the device which enables the system to be connected to wireless off-board services via access to voice/data networks. Currently, in most cases the cost of connection to these services is high. Therefore a user must be sure that only they are utilising this service, where and when they allow it.

3.1.1. Choice of Base Security Architecture

The security architecture developed in this thesis must be provably secure, which can be achieved by using trusted algorithms and methodologies. Moreover the developed architecture must be scalable and perform favourably in a networked environment. Finally it must provide a facility for the sharing of secrets between known devices in the network. As discussed in Section 2.13.4 there are two general methods for sharing a secret between two entities, public-key and symmetric-key cryptography.

In the case of this thesis the adopted methodology is required to execute on mobile devices of differing specifications. Moreover, all devices in the network are known. Therefore the level of security offered by the Diffie-Hellman is not needed as symmetric-key methods are suitable for use in this architecture. Moreover Ciampa (2005, p. 159) states that as the popularity of mobile devices increases, so to grows the need for security on these devices as to provide a facility to protect the device from a range of ever increasing attacks. Ciampa also states that these mobile devices are also easy targets for theft or loss because of their small size. This last point is on of the defining factors in the justification of the base chosen methodology. Public-key cryptography is used to generate and exchange secret keys across an unsecured channel. However, in most cases these keys are stored locally on the device.

Therefore if the device were lost or stolen the security of the entire system has been compromised. A solution to this is to generate a dynamic key for communication with each device in the network from a user known secret. As this key is dynamically generated it is not stored on the local mobile device. Therefore if a device were to be lost or stolen the system would not be compromised, as the attacker would not know the secret used to dynamically generate the key.

As such, the authentication mechanism developed in the current thesis for device authentication is based heavily on the SNMPv3 user security model (USM).

3.1.2. SNMPv3 USM

As outlined in Section 2.19 SNMP is a set of protocols for the management of network devices. This section concentrates on the user security model defined for SNMPv3 in RFC2754 (Blumenthal and Wijnen 1999). As defined in RFC2754 the major goals of the SNMP Security Model are to:

1. Provide verification of the integrity of each received message through the network;
2. Provide verification of the identity of the user on whose behalf an SNMP message has been generated;
3. Provide a method of detection for SNMP messages whose time generation is not recent.

The first goal of the SNMP Security Model is based around the SNMPv3 USM password-to-key algorithm. This algorithm enables a user to dynamically generate shared secret keys for each SNMP-enabled device in the network with only having to remember a single password. In the password-to-key algorithm, the user enters a password, which is appended to the end of the original password string until the string is 1 MB in length. The justification behind padding user passwords to 1MB is to drastically increase the time and computing power needed to perform an attack on the password. This padded (1 MB) string is passed through a secure hash function to produce an authKey. Next the authKey is localised using the key localisation algorithm. This algorithm takes the unique ID of the remote device (snmpEngineID) and encapsulates this string with the user's dynamically generated authKey. This

process is then fed through a secure hash function to produce a localised key, which is a shared secret between the user (who can generate the authKey) and the remote device, as the new key is localised with its unique ID.

3.2. Experiment 1: SNMPv3 Password to Key Algorithm

The first step in the design of a two-tier security architecture consists of an investigation into the ability of mobile devices with limited memory and processing power to use a modified version of the SNMPv3 password-to-key algorithm for dynamic key generation. This experiment tested the size that the user's password can be realistically padded to on mobile phones of differing specifications.

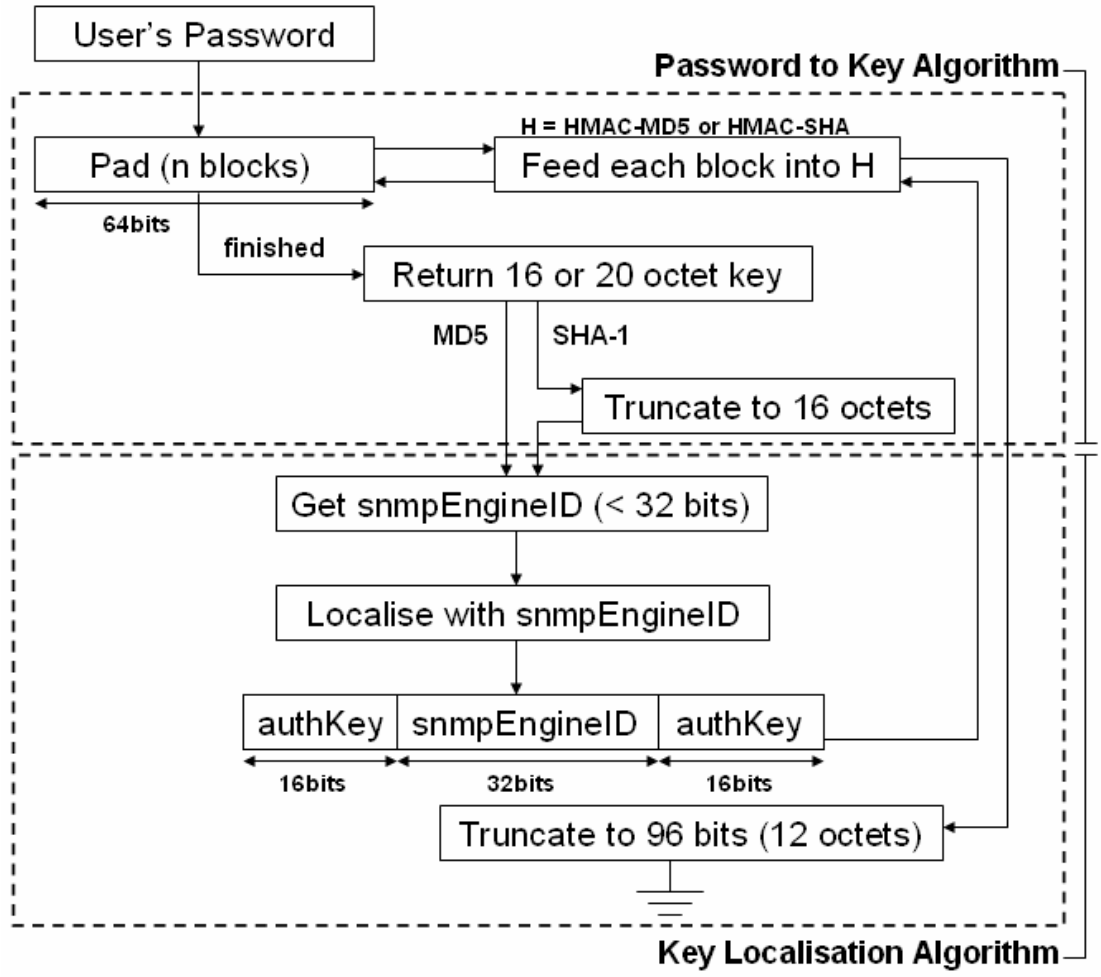


Figure 3.2. SNMPv3 Password to Key and Key Localisation Algorithms

Figure 3.2 outlines the steps taken in the first experiment. Firstly the user's password is padded to *n* number of 64bit blocks. Each 64bit block is fed into either

MD5 or SHA-1 HMAC secure hash algorithm. If the selected algorithm is HMAC-SHA then the 20octet (160bit) authKey is truncated to 16octets (128bits) as allowed by RFC2104 (Krawczyk, Bellare and Canetti 1997). However, if HMAC-MD5 was chosen then the authKey is already 16octets (128bits) in length. This concludes the SNMPv3 password-to-key algorithm.

The next step is to localise the key with a specific snmpEngine. This is achieved by retrieving the unique identifier for the snmpEngine (snmpEngineID) and encapsulating this with the generated authKey. If the snmpEngineID is less than 32bits in length then it is padded to 32bits. The encapsulated string is now of 64bits which is the HMAC block size used. Therefore the encapsulated string is fed into the original secure hash algorithm, and either a 20octet (160bit) or a 16octet (128bit) string is returned. Finally, the output localised key is truncated to 12octets (96bits) as allowed by RFC2104 (Krawczyk, Bellare and Canetti 1997).

3.3. Experiment 2: Mobile SNMPv3 USM over Bluetooth

This experiment extended the first experiment in order to test the viability of a mobile SNMPv3 implementation on mobile devices using Bluetooth as the networking medium. As the targeted mobile device is a mobile phone, power consumption becomes an issue when using the device to transmit data utilising a networking medium. However, as outlined in Section 2.10.1 Bluetooth is a wireless networking medium designed as a low powered cable replacement technology.

The first task in experiment 2 was to extend experiment 1 to create a slim-lined SNMPv3 protocol for use in a mobile environment on restricted devices, which satisfied the goals of the SNMPv3 USM as outlined in Section 3.1.2. In this case the device is the entity to be authenticated. Therefore the device itself requires a unique identifier in order for it to be tracked in the system. As the devices used in this thesis are GSM mobile phones a unique identifier is readily available – the International Mobile Equipment Identifier (IMEI)..

3.3.1. IMEI

The International Mobile Equipment Identifier (IMEI) is a unique 15 digit number configured at the time of manufacture in all GSM mobile devices. The IMEI

operates on the same principal as Media Access Control (MAC) addresses assigned to network interface cards which are installed in every desktop, laptop and server in the world. The 15 IMEI is split into sections including 2 digits which represent the vendor, so that each vendor can track allocated and available IMEI numbers.

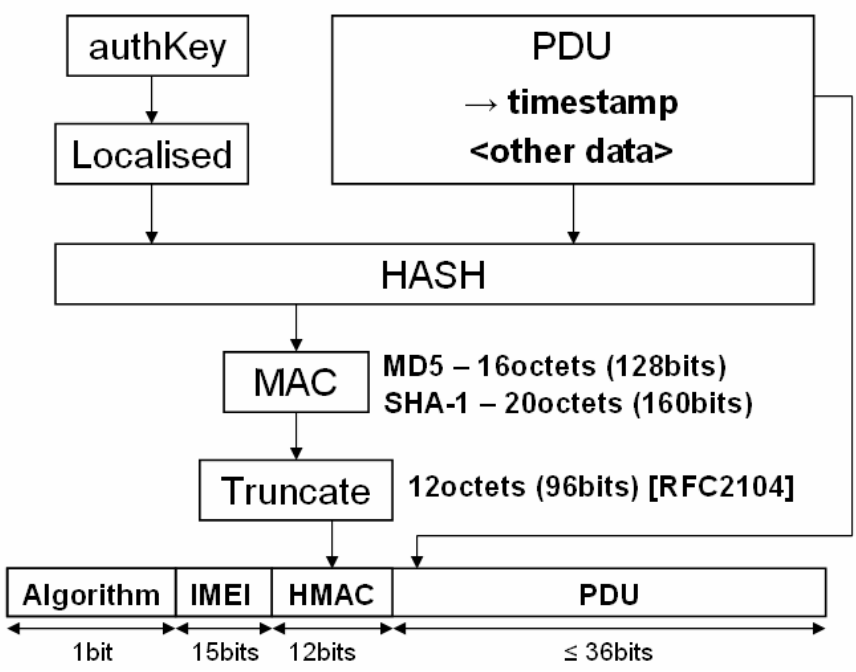


Figure 3.3. SNMPv3 Authentication Protocol

Figure 3.3 details the SNMPv3 Authentication Protocol used in experiment 2. This protocol extends experiment 1 by adding services to satisfy the timeliness requirement of SNMPv3 USM in order to circumvent replay attacks. In this case the client SNMP engine is required to keep track of the servers current time. This is simply achieved by selecting a discovered Bluetooth server device, asking the server the current time, then storing this timestamp and the time since this value was polled in the client device. When the protocol is transmitted the time since polling is added to the original server time and inserted into the protocol as the 'timestamp'. If the transmission arrives at the server with in a specified time window the timeliness requirement is satisfied. If the transmission arrives at the server with a time value outside this preset window the transmission is dropped. RFC2574 defines a timeliness window of 150 seconds (Blumenthal and Wijnen 1999). However this value is defined in a situation where transmission occurs across a potentially

convoluted public TCP/IP based network. The investigation of an appropriate timeliness window was the primary goal of experiment 2.

The protocol listed in Figure 3.3 dynamically generated the authKey as outlined in experiment 1. This key is localised with the snmpEngineID of the selected server. Next the timestamp and other data in the protocol data unit (PDU) is hashed using the localised key. The output of this function is a HMAC of either 16 or 20octets depending on the algorithm used, and is then truncated to 12octets as allowed by RFC2104 (Krawczyk, Bellare and Canetti 1997).

The final step is to construct the protocol string in the correct format for transmission. In this case the leading bit is used to represent the algorithm used (0 for HMAC-MD5, 1 for HMAC-SHA). The next 15 bits contain the deviceID which is simply the unique IMEI of the mobile device. Next the 12octet hashed protocol data unit (PDU) is inserted into the protocol string. Finally the original PDU is appended to the end of the protocol string in the same format as was used to create the hash. When the protocol string arrives at the server the timestamp is checked to evaluate whether the timeliness requirement has been satisfied. If the timeliness requirement is not satisfied then the packet is dropped. If it is satisfied, then the server segments the protocol string, and attempts to retrieve the localised key for the device as defined by the IMEI in the protocol string. If there server does not have a record of the deviceID then it will notify the device that it is an unknown. Next the server calculates its version of the hashed PDU and compares it to the hash in the original protocol string. If the values for both of the hashes are the same then the device is authenticated. If the hashes do not match the device is notified that a wrong password was entered.

3.4. Development Platform

The requirement for this protocol to run on mobile devices requires the use of a cross-platform development environment with a small footprint. It is this requirement that has lead to the system being implemented predominantly in Java 2 Micro Edition (J2ME) for the evaluation of both experiment 1 and experiment 2. A server environment will also be developed for experiment 2 in Java 2 Standard

Edition (J2SE). Moreover, an implementation in J2ME does not constrain the protocol to a subset of mobile devices, as J2ME is platform independent this implementation is cross-platform.

3.5. Java 2 Micro Edition

As J2ME implementations are designed to run on resource constrained mobile devices the footprint of the J2ME virtual machine must be kept to a minimum. However, there are a broad range of mobile devices on the market today and these devices have differing resources. For example most personal digital assistant (PDA) based devices can offer a greater range of services including processing power and memory, than that of a base level implementation on a limited mobile phone based device. It is because of this differing of capabilities that J2ME consists of different configurations.

3.5.1. J2ME Configurations

Keogh (2003, p. 12) defines two available J2ME configurations. Theses are the Connected Limited Device Configuration (CDLC) and the Connected Device Configuration (CDC). The CDLC is designed for use on small 16 or 32 bit computing devices with limited memory and battery life. The CLDC implements a trimmed version of the java virtual machine known as the KJava Virtual Machine or (KVM). CLDC is the J2ME configuration used in implementation of this thesis, on all three mobile devices.

Conversely CDC is designed for 32 bit devices with a minimum of 2 megabytes of memory, and can afford to implement a fully functioning java virtual machine Devices that utilise the CDC include everything from home appliances to high level smart phones.

3.5.2. J2ME Profiles

The use of one of the listed J2ME configuration also constrains the level of sophistication that can be programmed into the device. This is defined by the J2ME profiles. As mentioned previously, packages that involve high amounts of processing are only available to devices which implement the connected device configuration. Moreover the profiles available to devices implementing the

connected limited device configuration include the Mobile Information Device Profile (MIDP) which contains a set of base level classes that provide services for storage, user interface and wireless networking. MIDP is the profile utilised by this implementation.

3.6. Mobile Cryptography

As stated previously the implementation in this thesis uses the J2ME connected limited device configuration (CLDC) and utilises the mobile information device profile (MIDP). However, as discussed in Section 3.5.2 MIDP provides only a base level implementation of required packages. Therefore there is no implemented cryptography API in MIDP.

3.6.1. Bouncy Castle Cryptography API

A solution to this issue was to incorporate a third-party API. The chosen API was the Australian based java cryptography effort known as the Legion of The Bouncy Castle. The bouncy castle APIs form a complete java cryptography implementation, consisting of cryptographic APIs for J2EE, to a subset of APIs with a small footprint for use in J2ME (The Legion of The Bouncy Castle). Moreover, a high percentage of the cryptographic algorithms implemented are java ports from well known and trusted sources such as Schneier's Applied Cryptography (1996).

3.7. Bluetooth with J2ME

To compliment the lack of cryptographic APIs in J2ME, there is also a lack of base level support for Bluetooth. In order for java applications to access Bluetooth hardware on mobile devices these devices must implement the JSR-82 set of APIs for Bluetooth (Sun Microsystems 2002).

3.7.1. Third-Party Bluetooth APIs

The JSR-82 APIs provide functionality that can be used to manipulate the Bluetooth hardware on mobile devices. However, in this thesis third-party Bluetooth APIs were used to speed up the development lifecycle. The Bluetooth APIs used in this thesis were the BenHui (Hui) and Java Bluetooth (JavaBluetooth.org) APIs.

The Java Bluetooth stack was used in conjunction with the BenHui API to extend the client/server application that featured as an example in the BenHui APIs. This

example implementation was released under a general public license (GPL), and is therefore free from restriction as defined in GNU.org (1991).

3.8. Experiment 3: Evaluation of the Ang System

The third and final experiment undertaken by this thesis was to evaluate the face recognition system developed by Ang (2005) for use as either a user authentication, or user tracking system for the vehicle environment. The aim of the Ang system was to develop an identification mechanism using face recognition for a wireless network in order to provide user specific network services to identified users. The Ang system extended a free eigenfaces based (see Section 2.17.2) implementation of a 'quick and dirty' face recognition algorithm obtained from Sluggish Software (Sluggish Software). This system was then modified to operate in real time, with input from a complementary metal semiconductor sensor (CMOS) web camera.

The sets of images used for testing the Ang system were obtained by capturing images of enrolled users sitting in front of the camera and acting 'naturally'. Each set of ten images used different environmental variables, such as different backgrounds and differing head poses. However, both the camera and the chair which the enrolled users sat were both fixed at a distance of 70 centimeters apart. Ang conducted two major tests. In the first test users were asked to center their face in the capture view and change facial expression with out changing the orientation of their head, this test will be referred to as *ang_test1*. The second test allowed users to move their entire head around in the capture window, and will be referred to as *ang_test2*. Ang also tested the effect that different image types had on the accuracy of the system and concluded that colour images performed better than black and white images. The following section outlines the results obtained by testing the system using small (80 × 80) colour images.

3.8.1. Ang's Findings

The Ang system was tested with a total of 10 users enrolled in the system. The first experiment tested images enrolled and tested on a black background. The results as shown in Table 3.1 bellow are favorable, with a high accuracy rate.

Table 3.1. Face Recognition accuracy on black background (Ang 2005)

Black Background	
5 enrolled images ang_test1	100%
5 enrolled images ang_test2	90%
10 enrolled images ang_test2	100%

The second test was conducted using enrolled images captured on a white background, where testing was also conducted on a white background. This test was conducted to determine if backgrounds of different brightness would affect the accuracy of the system.

Table 3.2. Face Recognition accuracy on white background (Ang 2005)

White Background	
5 enrolled images ang_test1	98%
5 enrolled images ang_test2	94%
10 enrolled images ang_test2	100%

The results as outlined in Table 3.2 again present positive results as to the accuracy of the Ang system. A final test is to examine the effect of differing backgrounds on the level of accuracy of the system. The following table outlines the findings of a test where the enrolled images include a white background. Conversely the testing images have differing backgrounds.

Table 3.3. Face Recognition accuracy on different backgrounds (Ang 2005)

Differing Background	
5 enrolled images ang_test1	46%
5 enrolled images ang_test2	56%
10 enrolled images ang_test2	42%

The findings outlined in Table 3.3 present results which are on average less than 50 percent accurate. Therefore it can be safely concluded that this system performs poorly when tested with images which consist of a greatly different background to that of the enrolled images. However the Ang system performed favorably when the images used for testing had the same background as the enrolled images. Moreover, both Table 3.1 and 3.2 present an accuracy of 100 percent when 10 images are enrolled as apposed to a 90 and 94 percent accuracy respectively for the same test with only 5 images enrolled.

3.8.2. In-vehicle Face Recognition

The aim of this experiment is to evaluate the Ang face recognition system as a method for user authentication, and or user tracking. The requirement for user authentication in a vehicle environment is to be used in parallel with the mobile SNMPv3 protocol outlined in Section 3.3 for mobile device authentication. This addition of this system would employ an added level of security, as the system could verify a user's identity and based on this identity, the users allowance to use an authenticated device. Moreover the addition of this system could act as a user tracking system which could be used for session monitoring, as in any security architecture a successful method to log a user off the system, is as integral to system security as a trusted authentication method. In this situation the system would routinely monitor the vehicle environment for authenticated users.

In order to emulate the conditions of the vehicle environment the CMOS camera was mounted at a position slightly above the user's eye line as to emulate mounting on a vehicle sun visor. Moreover, during the process of user enrolment users were required to vary their distance to the camera, including both head distance and chair distance. This was done to emulate the varying seat position in the vehicle. The background in both the testing and enrollment images was consistent yet not a flat contrasting shade as was used during testing the Ang system, as to emulate the standard yet inconsistent background of the vehicle environment due to ever changing lighting conditions. The non-standardised background should not be an issue when evaluating the Ang system, as in Section 3.8.1 it was concluded that testing images which had the same background as the enrollment images returned positive results.

The aim of this thesis is to evaluate the Ang system with 1 to 10 enrolled users all of which have 10 enrolled images. The first test enrolled users who were the least alike in appearance and cultural background as to evaluate the performance of the system in a best case scenario. The second test will perform the inverse of the first test in that enrolled users will be enrolled which are similar in appearance, this test will evaluate the performance of the system in a worst case scenario.

4

4. Discussion and Results

The following chapter examines the results which were attained in implementing both experiments one and two as outlined in chapter 3 on three mobile devices all with significantly different specifications.

4.1. Mobile Phones Used for Testing

The mobile phones used to test both experiment one and two, are vastly different in their specification, including memory and processing power, and level of java support. Moreover, two of the three devices are Bluetooth enabled however only the Nokia 6600 implements the JSR-82 Bluetooth for Java API, and is therefore the only device where any java based Bluetooth experiments can be tested. All three devices connect to the GSM mobile voice based network and connect to GPRS mobile network for data services.

4.1.1. Nokia 6610

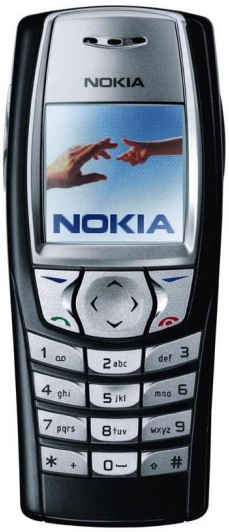
Released in the fourth quarter of 2002 the Nokia 6610 was a revolutionary device upon release. This device was one of the first in its class to feature a bright colour screen, java support and other advanced capabilities.

The Nokia 6610 is the base level device that was used in testing. The base level java support in this device allows this device to form the ‘worst case’ scenario from a runtime perspective.

Specifications

Operating System: Nokia Series 40 OS

Memory: 725 kB



Network: Tri-band GSM (900/1800/1900)

Data Connectivity: GPRS

Bluetooth: No

Screen Resolution: 128×96

Java Support:

- **Profiles:** MIDP-1.0
- **Configuration:** CLDC-1.0
- **Bluetooth (JSR-82):** No
- **Virtual Processor Speed:** 1.2 MHz
- **Max Jar Size:** 64 kB

Figure 4.1. Nokia 6610

4.1.2. Nokia 6600

The Nokia 6600 is the mid-level device. However it is used exclusively in testing experiment 2 as it is the only phone out of the test devices which includes the java Bluetooth API.

Specifications

Operating System: Symbian Series 60 OS 7.0s

Memory: 6 MB

Network: Tri-band GSM (900/1800/1900)

Data Connectivity: GPRS

Bluetooth: Yes

Screen Resolution: 176×144

Java Support:

- **Profiles:** MIDP-2.0
- **Configuration:** CLDC-1.0
- **Bluetooth (JSR-82):** Yes
- **Max Jar Size:** Dynamic

Figure 4.2. Nokia 6600

4.1.3. iMate SP3i

The iMate SP3i is the most advanced device tested. This device is the realisation of a convergence as it contains a full featured mobile phone, coupled with an advanced Microsoft based PDA.

Specifications

Operating System: Microsoft Pocket PC 2003 SE OS

Memory: 32 MB

Processor: TI OMAP 730 processor

Network: Tri-band GSM (900/1800/1900)

Data Connectivity: GPRS (Class 10 (4+1/3+2 slots))

Bluetooth: Yes

Screen Resolution: 176×220

Java Support:

- **Profiles:** MIDP-1.0/2.0
- **Configuration:** CLDC-1.0
- **Bluetooth (JSR-82):** No

Figure 4.3. iMate SP3i

4.2. Experiment 1: SNMPv3 Password to Key Results

The aim of this experiment as outlined in Section 3.2, was to investigated the size that an input password could be effectively padded to in accordance with the SNMPv3 User Security Model (Blumenthal and Wijnen 1999). The password-to-key algorithm was tested with a fixed password and snmpEngindID for key localization. This enabled the results to be compared as all devices were undertaking the same tasks. The runtimes were gathered and graphed for each device calculating to the same block sizes using both HMAC-MD5-96 and HMAC-SHA-96 algorithms over a nominated range of block sizes, up to a maximum of 1 MB as defined in the SNMPv3 USM specification (Blumenthal and Wijnen 1999).

4.2.1. Results for HMAC-MD5-96

Figure 4.4 outlines the runtime of the average of 5 trials executed on all three devices using the HMAC-MD5-96 algorithm to pad to *n* blocks of 64bits.

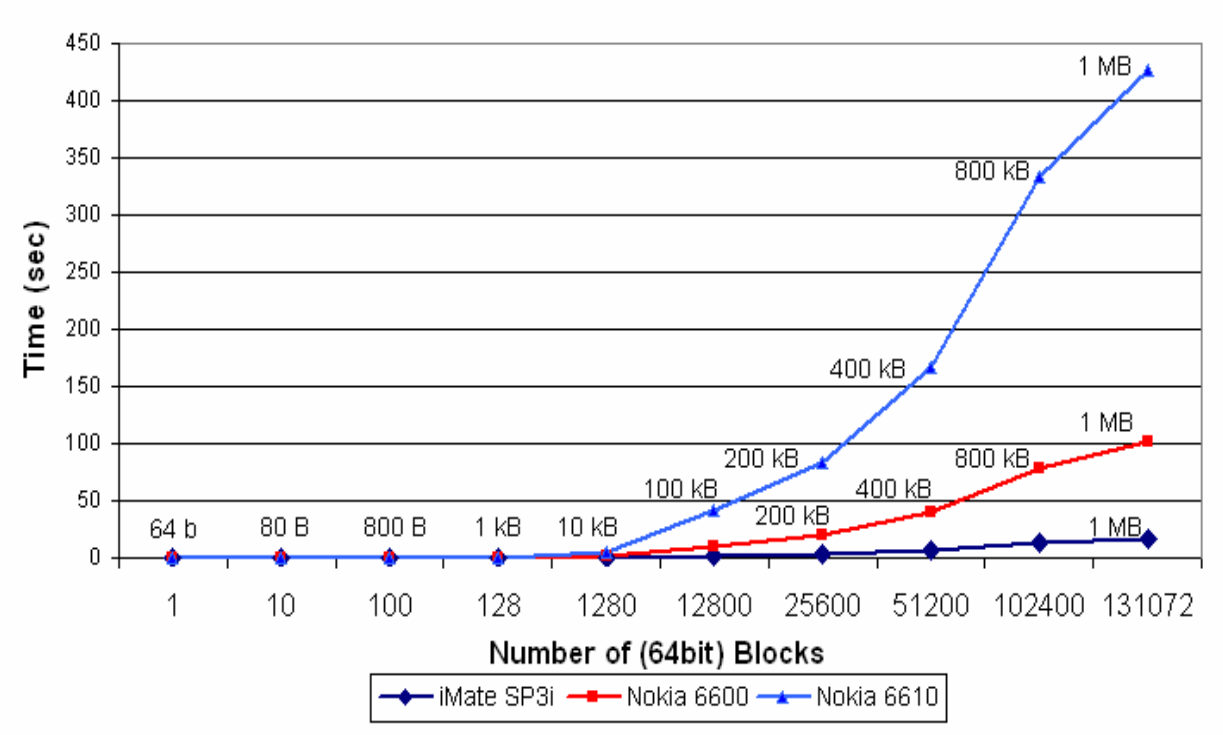


Figure 4.4. Average runtime of the HMAC-MD5-96 algorithm

From the figure above, it is interesting to note the variance in runtimes between the devices, namely the Nokia 6610. A possible cause for such a varied result of a runtime of 7:06 minutes to pad the password to 1MB on the Nokia 6610, as compared to runtimes of 1:41 minutes and 17.29 seconds to pad the password to 1MB for the Nokia 6600 and iMate SP3i respectively, is the shear lack of available memory on the Nokia 6610. Moreover, Figure 4.4 highlights that a block size of 1280 blocks (10 kB) is the maximum a password can be efficiently padded using the HMAC-MD5-96 algorithm on all three devices, before the runtime increases to an impractical level.

4.2.2. Results for HMAC-SHA-96

Figure 4.5. illustrates the average runtime after 5 trials of the HMAC-SHA-96 algorithm padded to *n* blocks of 64bits on the three mobile devices used for testing.

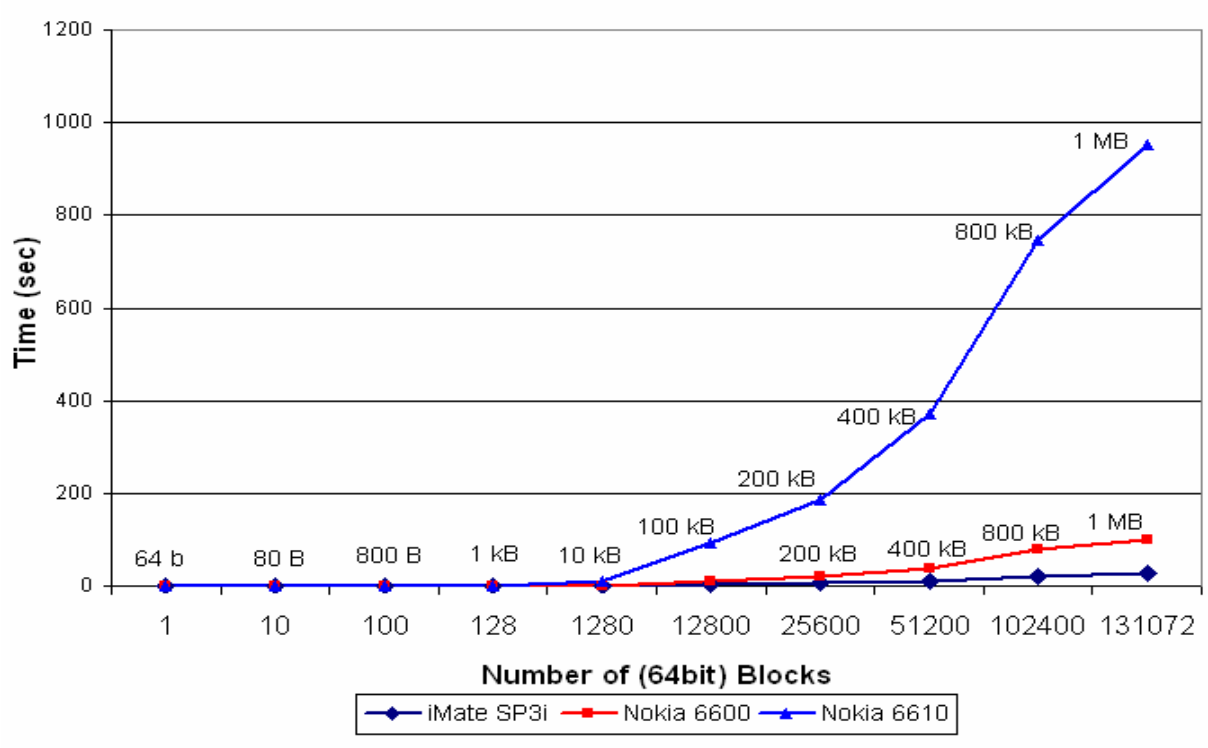


Figure 4.5. Average runtime for the HMAC-SHA-96 algorithm

The above figure illustrates the average runtimes of the HMAC-SHA-96 algorithm as tested on all three mobile devices. These results are consistent with those discovered in section 4.2.1. However, in these results the difference is emphasised. The variance in runtimes between the Nokia 6610 and the other devices proves the authors assertion that the devices performance is restricted due to an insufficient availability of memory. Figure 4.5 highlights a runtime of 15:53 minutes for the Nokia 6610 to pad the password to 1MB. This is drastically contrasted by the Nokia 6600 and the iMate SP3i which took 1:40 minutes and 26.24 seconds respectively. It is interesting to note the level of increase in runtime between the two algorithms implemented. In this case the runtime of the Nokia 6610 increased by over 100 percent, where the difference in runtime between the other devices was minimal at best. However, the results illustrated in Figure 4.5 mirror that of Section 4.2.1, in that the maximum size a password can be efficiently padded to using the HMAC-SHA-96 algorithm is 1280 blocks (10 kB).

4.2.3. HMAC-MD5-96 versus HMAC-SHA-96

Figure 4.6 and the accompanying Table 4.1 bellow provide a comparison of the runtimes of both the HMAC-MD5-96 and HMAC-SHA-96 algorithms on the three testing devices. The table and accompanying figure below highlight the difference in runtime between the devices for both algorithms.

Table 4.1. Runtime Comparison between HMAC-MD5-96 and HMAC-SHA-96

Device and Algorithm	Number of (64bit) Blocks									
	1	10	100	128	1280	12800	25600	51200	102400	131072
iMate SP3i MD5	0.1	0.1	0.1	0.1	0.3	1.8	3.4	6.7	13.5	17.3
Nokia 6600 MD5	0.3	0.3	0.4	0.5	1.4	10.1	20.1	39.9	79.2	101.6
Nokia 6610 MD5	0.2	0.2	0.5	0.6	4.3	41.8	83.4	166.8	333.8	426.1
iMate SP3i SHA	0.1	0.1	0.1	0.1	0.4	2.7	5.3	10.3	20.7	26.2
Nokia 6600 SHA	0.3	0.3	0.4	0.5	1.3	9.8	19.4	39.3	78.3	100.6
Nokia 6610 SHA	0.2	0.3	0.9	1.1	9.5	93.5	186.5	372.3	743.0	953.2

----------Runtime in Seconds----------

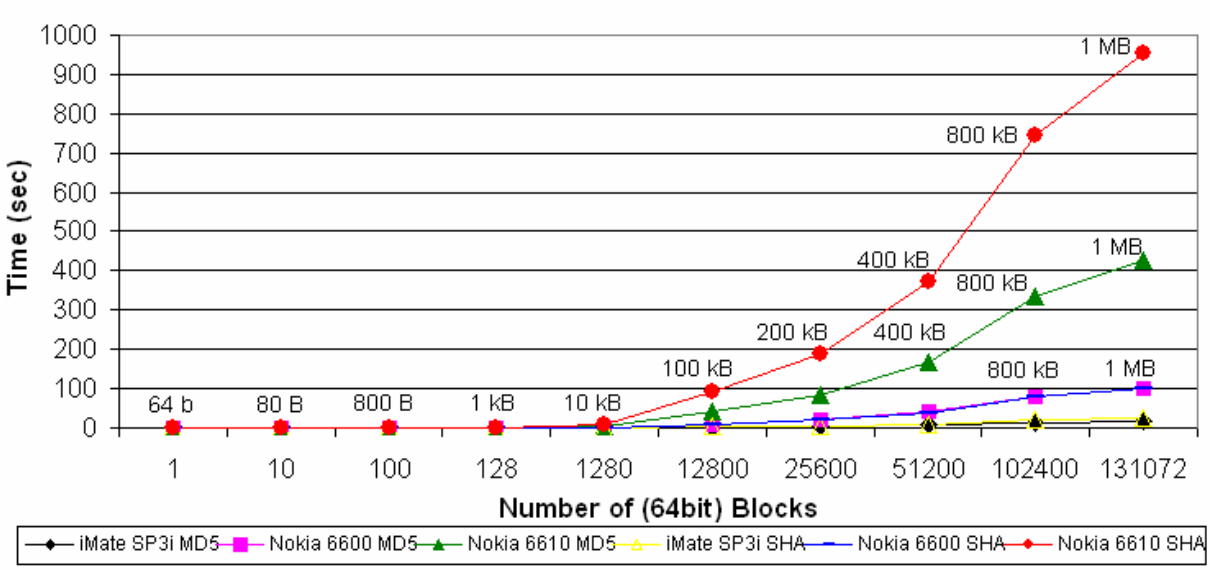


Figure 4.6. Runtime Comparison of HMAC-MD5-96 and HMAC-SHA-96

The above table and figure clarify the difference in efficiency between the devices. In this case we can see a marginal increase in the runtime cost of HMAC-SHA-96 over HMAC-MD5-96 in both the Nokia 6600 and the iMate SP3i. These results emphasise the discrepancy in runtimes between the two algorithms on the Nokia 6610.

Figure 4.6 also strengthens the assumption that the maximum block size that a user's password can be efficiently padded to on all devices is 1280 blocks (10 kB). Table 4.1 shows an average runtime on the Nokia 6600 and iMate SP3i of less than 1.5 seconds for password-to-key and key localization.

4.3. Experiment 2: Bluetooth Mobile SNMPv3 USM Results

As outlined in Section 3.3 the aim of experiment 2 was to extend the operation of the algorithm used in experiment 1 in order to develop a complete protocol for SNMPv3 based authentication across a Bluetooth network. The goal of this experiment was to discover the size of the time window necessary to implement the timeliness requirement of the SNMPv3 user security model over Bluetooth. The realisation of this protocol consisted of a Bluetooth client and server application developed in J2ME and J2SE.

The timeliness requirement is necessary in order to prevent, or at least decrease the protocols susceptibility to message stream modification attacks. These attacks can include the unauthorised modification, deletion and replay of legitimate traffic as defined in Section 2.18.2.

The client and server application was implemented in both J2ME and J2SE. This implementation was based on an example application which accompanied the BenHui Bluetooth API as described in Section 3.7.1. As this implementation required interfacing with Bluetooth hardware in J2ME the mobile device running the application was required to implement the JSR-82 Java Bluetooth. This restriction proved a major issue as only one of the testing devices contained a JSR-82 implementation, this was the Nokia 6600. Therefore the results for experiment 2 we obtained solely from testing the software on the Nokia 6600.

- Start the server on the PC to listen for incoming Bluetooth connections
- Start the client MIDlet on the mobile device (Nokia 6600)
- Perform Bluetooth device discovery
- User selects a device (server) to connect to (key localisation)
- Client asks the server for the current time

- Server replies and send the current time to the client
- Next the user is prompted to entire their password for the selected device
- Program executes the protocol (See Figure 3.3)
- Client send the protocol string to the server
- Server looks at the timestamp
- If the timestamp is not within the specified window packet is dropped
- Else the server computes a hash of the PDU and compares it with the one sent by the client.
- If they match the device is authenticated

Figure 4.7. Experiment 2 program flow

Figure 4.7 above outlines the basic flow of the experiment 2 implementation. As stated the aims of this experiment are to determine a suitable time window for the developed protocol.

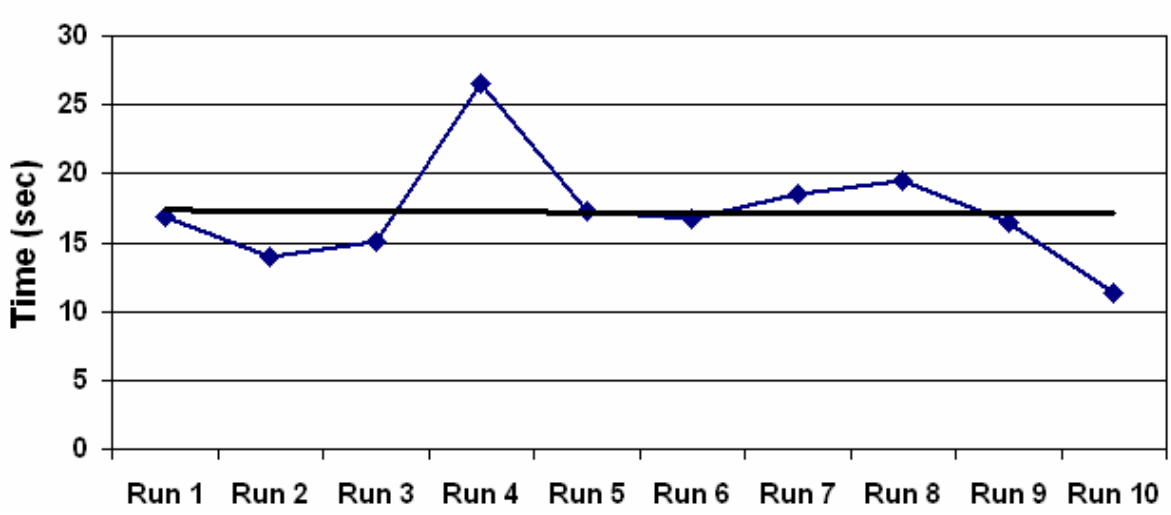


Figure 4.8. Execution time for protocol over Bluetooth

Figure 4.8 outlines the total runtime of the entire protocol developed for experiment 2. However this runtime does not include device discovery. This is due to the fact that the process of Bluetooth device discovery is highly varied, and can be influenced by such factors as number of devices in range and the distance to these devices. Therefore the calculated necessary time window would be heavily

dependent on the speed of the device discovery. Due to this fact the time window calculation is started as soon as the client selects a Bluetooth device to connect to. Figure 4.8 above illustrates an average of 17seconds for the completion of the protocol over Bluetooth. Therefore a time window of approximately 30 seconds should be sufficient to protect against message stream modification attack.

4.4. Experiment 3 Results

As discussed in Section 3.8, the third experiment was to evaluate the Ang face recognition system for use as a user authentication and/or user tracking mechanism for the vehicle environment.

The Ang system was tested with different numbers of enrolled users, from 2 users to 10 users. The number of enrolments was restricted due to the assumption that in a shared device environment a single device would not be shared among more than 10 users.

After 10 users were photographed with the web camera batch testing cases were compiled with differing number of enrolled users from 2 to 10. In the first test users were added to the system that had the most differing appearance, sex and cultural background to that of the test case. This was done to form an initial best case judgment of the success of the system.

Testing was carried out in a way that would emulate the vehicle environment. Because of this there was no flat white/black background as in the prior research. Moreover, the users were asked during the initial enrolment period to move there position to the camera as to emulate the variance of a car seat in differing positions. The users were asked at the time of initial enrollment to capture a large number of images, which were later restricted to 10 images per user. These 10 images were selected based on their orientation and distance from the camera, so as to provide a varied, yet consistent set of images per user.

4.4.1. Test 1: Best Case

The results from the first test proved rather promising. As shown in the figure bellow.

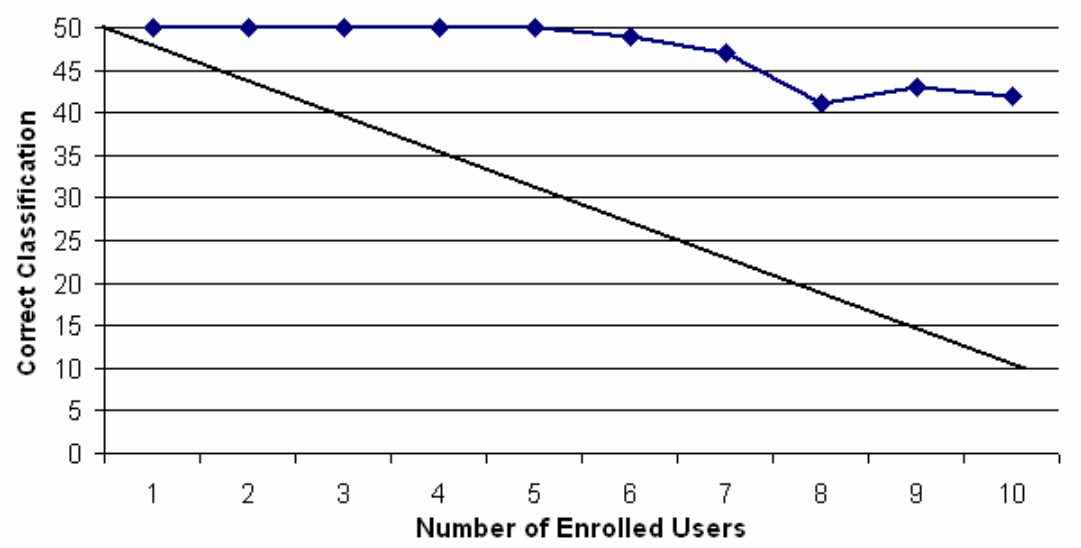


Figure 4.7. Results test 1: Best Case

Figure 4.7 illustrates the accuracy of the Ang system in the best case scenario. The trend line is projected onto the above figure as it reflects the chance performance of the system in that when one user is enrolled, there is a 100 percent chance that this user will be selected. Conversely when 10 users are enrolled the system the chance performance drops to 10 percent.

4.4.2. Test 2: Worst Case

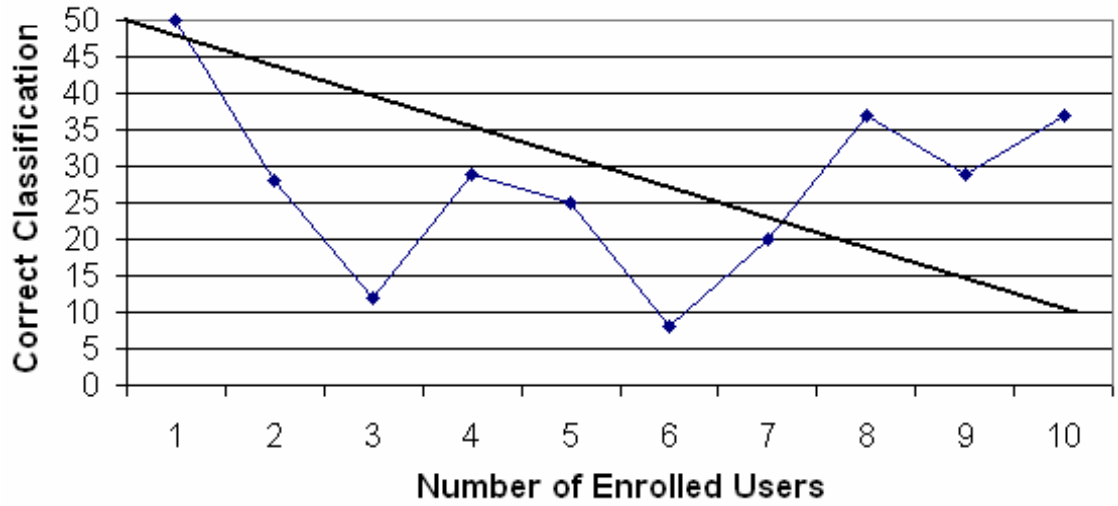


Figure 4.8. Results test 2: Worst Case

Figure 4.8 outlines the results on testing the Ang system with enrolled user of similar facial characteristics. The trend line shows that the system performed worst than would be expected from a chance performance.

It was noticed during test 1 of the face recognition that wrong classifications were occurring mostly with users whose username was alphabetically before the test case. It was this finding that lead to test 3.

4.4.3. Test 3: Modified Best Case

Test 3 consisted on the same enrolment methodology as test 1. However the username of the test case was changed to "zz", in order to test the assumption that the system was running through the enrolled users in alphabetical order until it found a user that was above a hard coded threshold. Test 3 provided the following results.

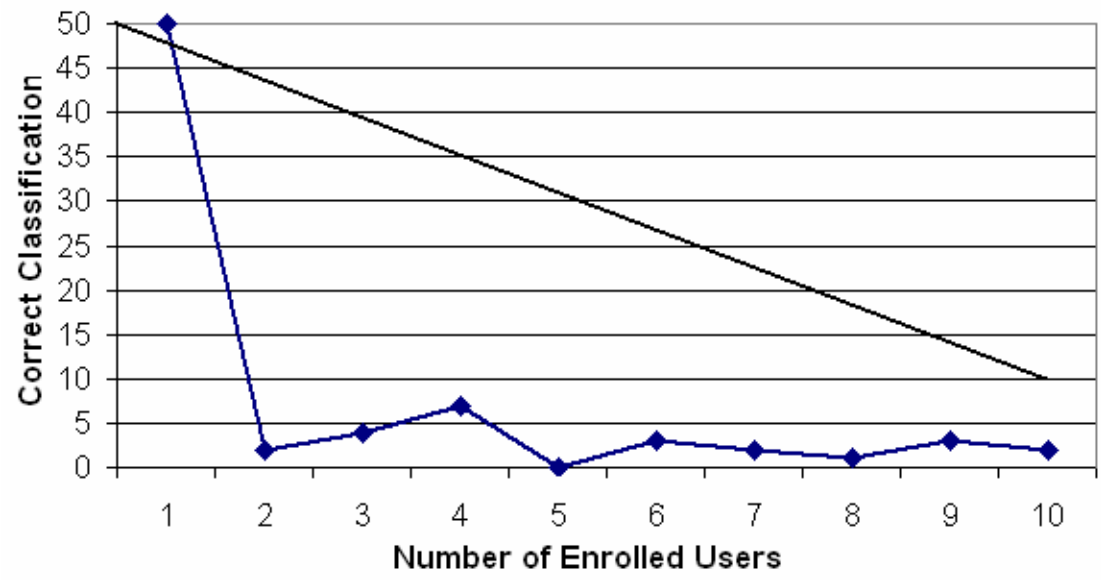


Figure 4.9. Results test 3: Modified Best Case

Figure 4.9 outlines the results obtained for test 3. In this case the system performed drastically worse than would be expected from a change performance.

The results for test 3 with the same user set seemed to agree with the previous assumptions of the system. Moreover, the system always returns a user as authenticated. It either returns the closest or the alphabetically closest to the

presented image. Because of this the performance of the system must be able to assure that it will identify the correct user. Otherwise its use as an authentication or verification of identity mechanism is not sufficient to be applied to a security application.

4.4.4. Deceiving the Face Recognition System

Further testing of the Ang system discovered that it was easy to deceive the system. The first test was to access the ability to deceive the system by using a printed image of an enrolled user. The results of this test classified the printed image more accurately than it did the real entity. Next the system was tested by holding a picture of the back of a persons head to the camera and in this case a user was returned although with varied results. The main conclusion that can be made regarding the Ang system is that it is not accurate enough for use in a security architecture. Moreover, the system always returns a user as identified, even if the entity attempting to authenticate them self is not enrolled in the system, it will accept their appearance as an enrolled user's identity.

5

5. Conclusions and Further Work

The research within this thesis analysed the current situation of the automotive telematics industry, including the current design goals for vehicle based systems and most importantly the lack of adopted security protocols for the mobile platform.

This thesis proposed and implemented a security protocol for the vehicle paradigm based on the simple network management protocol (SNMP) version 3 user security model (USM). This protocol is the most widely adopted network management protocol for TCP/IP based networks. SNMPv3 was chosen over other security protocols as it utilised the services of symmetric-key cryptography, which unlike the public-key methods provides a protocol which is efficient to implement in software, scalable and that is proven to be secure by employing the services of mathematically proven cryptographic algorithms.

This security protocol was developed to provide a method for secure mobile device integration for automotive telematics based on two scenarios. The first being a shared vehicle origination where all staff are provided with a Bluetooth enabled mobile device. The security requirement for this scenario is that user preferences are stored on the mobile device and can be uploaded to any of the shared vehicles. The preferences stored on these devices can contain seat and steering wheel position, and even radio presets. The second scenario defined an example topology for the emulation of high level telematics systems using only consumer level devices. This emulation requires the use of a telematics control unit (TCU) as defined in Section 2.8.2. The TCU acts as a gateway between the consumer level devices and the onboard devices. Moreover, any vehicle based system should be primarily focused on safety. By employing the services of a workload manager as defined in Section

2.6.1 the current driving context can be analysed and information can be either provided to the driver in a standard or non standard way depending on the inferred context. Moreover an advanced method of interaction is required for the vehicle environment. This study has shown that voice based interaction is the most suited modality for the vehicle paradigm.

This thesis incorporated three major goals. First, an evaluation of the length a user's password could be efficiently mapped to on mobile devices of differing specifications, as a mobile implementation of the SNMPv3 USM. It was concluded from this study that a pad size of 10kB will provide adequate security, while remaining efficient on the three mobile devices used for testing.

Second, the first goal was expanded into a protocol for mobile device integration. This protocol was a full implementation of the SNMPv3 USM for device authentication using Bluetooth for the transmission between the client and the server. The objective of this goal was to evaluate the length of the time window required in order to protect the system from message stream modification attacks. It was concluded that it was not practical to incorporate Bluetooth device discovery in to the time window a the time taken for Bluetooth discovery is dependent on many factors including the number of, and distance to other Bluetooth devices in range. However, it was concluded that a time window of approximately 30 seconds should be sufficient to provide the transmission of slightly delayed legitimate messages, where at the same time be short enough for any attack to be rendered harmless.

Third, the Ang face recognition system was evaluated for use in a vehicle environment, although with varying results. Initially the results returned were favorable, as the first test was the best case scenario. The second test enrolled users were in order of their likeness in appearance to users already enrolled in the system. This was done to evaluate a possible worst case scenario. The results from this test were not as favorable as that of the first. In that the system inaccurately classified enrolled users at a rate less than the probable chance of the system. An anomaly was found while testing the Ang system. It was discovered that a high percentage of the wrong classifications were returning users with a username alphabetically before that of the test subject. In order to test this assumption the best case scenario of test

one was reevaluated. However in this case the test cases username was changes to "zz". The results for experiment 3 showed a correct classification level drastically lower than that of probable chance. Therefore it was concluded that as an authentication or user tracking mechanism this system was not accurate enough to be used for this purpose.

5.1. Further Work

As it has been shown that a method of secure device integration can be developed for use on mobile devices operated in a networked environment, the field of mobile security will be an integral industry in the furthering of security in all aspects of mobile device integration, from the vehicle, to mobile commerce applications where security is paramount.

As it has been shown that the Ang face recognition system is not sufficient to provide an adequate level of security to the application of mobile device integration, other more secure methods for transparent or limited interaction authentication, which may include the use of wireless radiofrequency identification (RFID) tokens. Conversely a higher level face recognition system may be the answer.

Moreover, it would be interesting to test the viability of the implemented system with transparent authentication. In this case the user would not have to physically enter their password prior to performing Bluetooth discovery. Extending this, possible methods for user tracking could be developed in order to track users' movements through out the use of the system, and could facilitate a method for secure sign off.

Another field of possible extension would be to intercept and study the Bluetooth communication between the client and server at time of the transfer of the developed protocol. This would be interesting to determine the level of possible added security if the Bluetooth communications link was encrypted.

6

6. References

Ailisto, H, Lindholm, M, Mäkelä, S-M and Vildjiounaite, E 2004, 'Unobtrusive user identification with light biometrics', paper presented to Proceedings of the third Nordic conference on Human-computer interaction, Tampere, Finland.

Alm and Nilsson 1995, 'The effects of a mobile telephone task on driver behaviour in a car following situation', *Accident Analysis and Prevention*, vol. 27, no. 5.

Alpern, M and Minardo, K 2003, 'Developing a car gesture interface for use as a secondary task', in *CHI '03 extended abstracts on Human factors in computing systems*, ACM Press, Ft. Lauderdale, Florida, USA, pp. 932-3.

Ang, SH 2005, 'Smart Noticeboard', Honours thesis, University of Tasmania.

Blumenthal, U and Wijnen, B 1999, *RFC2574: User-based Security Model (USM) for version 3 of the Simple Network Managment Protocol (SNMPv3)*.

Brick, J 1996, 'Intoxicated Driving', *Intoxikon International*.

Bühler, D, Vignier, S, Heisterkamp, P and Minker, W 2003, 'Safety and operating issues for mobile human-machine interfaces', paper presented to Proceedings of the 8th international conference on Intelligent user interfaces, Miami, Florida, USA.

Chiasserini, FC and Rao, RR 2000, 'Coexistence between Bluetooth and IEEE 802.11 CCK: Solutions to avoid mutual interference', *IEEE P802.11 Working Group Contribution*, vol. IEEE P802.11, no. 00/162r0.

Chou, W 2003, *Elliptic Curve Cryptography and Its Applications to Mobile Devices*, University of Maryland, College Park.

Ciampa, M 2005, *Security+ Guide To Network Security Fundamentals*, Second edn, Thomson Course Technology.

economist.com 2005, *The talking cure*, <http://www.cmu.edu/clips/v34.html>.

Fortunati, L 2001, 'The Mobile Phone: An Identity on the Move', *Personal Ubiquitous Comput.*, vol. 5, no. 2, pp. 85-98.

Fuchs, A and Spaur, C 2004, *Plug and Play Personal Telematics*.

gnu.org 1991, *GNU General Public License*, viewed Tuesday November 1 2005, <http://www.gnu.org/copyleft/gpl.html>.

Goodman, MF, Bents, FD, Tijerina, L, Wierwille, W, Lerner, N and Benel, D 1997, 'An Investigation of the Safety Implications of Wireless Communications in Vehicles'.

Green, P 2004, *Driver Distraction, Telematics Design, and Workload Managers: Safety Issues and Solution*.

Hahn, RW, Tetlock, PC and Burnett, JK 2000, 'Should you be able to use your cellular phone while driving', *Regulation*, vol. 23, no. 3, pp. 46-55.

Heisterkamp, P 2000, *Linguatronic Product-Level Speech System for Mercedes-Benz Cars*.

Heuvel, Hvd, Boudy, J, Comeyne, R, Euler, S, Moreno, A and Richard, G 1999, *The SpeechDat-Car Multilingual Speech Databases For In-Car Applications: Some First Validation Results*.

Hinckley, K and Horvitz, E 2001, 'Toward more sensitive mobile phones', paper presented to Proceedings of the 14th annual ACM symposium on User interface software and technology, Orlando, Florida.

Hopkins, B and Antony, R 2003, *Bluetooth for Java*, Apress.

Hui, B *Benhui.net Source for J2ME Bluetooth Mobile 3D MIDP 2.0*, viewed Tuesday November 1 2005, <http://benhui.net/>.

instat.com 2003, <http://www.instat.com/>.

Jakobsson, M and Pointcheval, D 2002 'Mutual Authentication for Low-Power Mobile Devices ', in *Proceedings of the 5th International Conference on Financial Cryptography* Springer-Verlag, pp. 178-95

JavaBluetooth.org *JavaBluetooth Stack*, viewed Tuesday November 1 2005, <http://www.javabluetooth.org/>.

Jöst, M, Häußler , J, Merdes, M and Malaka, R 2005, 'Multimodal interaction for pedestrians: an evaluation study', paper presented to IUI '05: Proceedings of the 10th international conference on Intelligent user interfaces.

Juliussen, E 2003, *The Future of Automotive Telematics*.

Keogh, JE 2003, *J2ME : the complete reference*, McGraw-Hill/Osborne, New York ; London.

Krawczyk, H, Bellare, M and Canetti, R 1997, *RFC2104: HMAC: Keyed-Hashing for Message Authentication*.

Malaka, R, Haeussler, J and Aras, H 2004, 'SmartKom mobile: intelligent ubiquitous user interaction', in *IUI '04: Proceedings of the 9th international conference on Intelligent user interface*, ACM Press, pp. 310--2.

Manalavan, P, Samar, A, Schneider, M, Kiesler, S and Siewiorek, D 2002, 'In-car cell phone use: mitigating risk by signaling remote callers', paper presented to CHI '02 extended abstracts on Human factors in computing systems, Minneapolis, Minnesota, USA.

Matthews, T, Dey, AK, Mankoff, J, Carter, S and Rattenbury, T 2004, 'A toolkit for managing user attention in peripheral displays', paper presented to Proceedings of the 17th annual ACM symposium on User interface software and technology, Santa Fe, NM, USA.

Menezes, AJ, Van Oorschot, PC and Vanstone, SA 1997, *Handbook of applied cryptography*, CRC Press, Boca Raton.

Microsoft 2005, *Microsoft Windows Automotive Conference 2005 Keynote*, viewed Thursday, 29 September 2005, <http://www.microsoft.com/japan/seminar/mwac2005/keynote/play.asp>.

Motorola 2005, *1930s Motorola History Highlights*, viewed Sunday September 25, 2005 <http://www.motorola.com/content/0,,117-282,00.html>.

National Conference of State Legislatures 2003, 'Appendix B. Existing State Laws Regarding Mobile Phone Use While Driving'.

Nora, S and Minc, A 1968, *L'informatisation de la société*, The Informatisation of Society, MIT Press, Boston, MA., Paris.

Parkes 1991, *Drivers Business Decision-making ability Whilst Using Carphones*, HUSAT Research Centre, UK.

Pfleeger, CP and Pfleeger, SL 2003, *Security in computing*, 3rd Int edn, Prentice Hall PTR, Upper Saddle River, N.J.

Queensland Business Review 2003, *Mobile Phone Penalties*, viewed 11 of July 2005, <http://www.qbr.com.au/index.cfm?storyid=17291&cp=displaystory.cfm>.

Redelmeier, DA and Tibshirani, RJ 1997, 'Association between cellular-telephone calls and motor vehicle collisions', *New England Journal of Medicine.*, no. 336, pp. 453-8.

Reithinger, N, Alexandersson, J, Becker, T, Blocher, A, Engel, R, Löckelt, M, Müller, J, Pfleger, N, Poller, P, Streit, M and Tschernomas, V 2003, 'SmartKom: adaptive and flexible multimodal access to multiple applications', paper presented to Proceedings of the 5th international conference on Multimodal interfaces, Vancouver, British Columbia, Canada.

Royal Society for the Prevention of Accidents (RoSPA) 2001, *The Risk of Using a Mobile Phone While Driving*.

Schneier, B 1996, *Applied cryptography : protocols, algorithms and source code in C*, 2nd edn, Wiley, New York.

Schneier, B 2004, 'Sensible Authentication', *Queue*, vol. 1, no. 10, pp. 74-8.

Schneier, B and Ferguson, N 2003, *Practical Cryptography*, Wiley.

Sluggish Software *Sluggish Software*, viewed Tuesday November 1 2005, <http://www.fuzzgun.btinternet.co.uk/Downloads.htm>.

Sodhi, M, Reimer, B, Cohen, JL, Vastenburg, E, Kaars, R and Kirschenbaum, S 2002, 'On-road driver eye movement tracking using head-mounted devices', paper presented to Proceedings of the symposium on Eye tracking research \& applications, New Orleans, Louisiana.

Stallings, W 1998, 'SNMPv3: A Security Enhancement For SNMP', *IEEE Communications Surveys*, vol. 1, no. 1.

Stallings, W 2003a, *Network security essentials : applications and standards*, 2nd edn, Pearson Education, Upper Saddle River, NJ.

Stallings, W 2003b, *Cryptography and Network Security*, Prentice Hall.

State Government of Tasmania 1999, *Traffic (Road Rules) Regulations: Section 300. Use of hand-held mobile phones*.

Strayer, DL and Johnston, WA 2001, 'Driven to distraction: Dual-task studies of simulated driving and conversing on a cellular telephone', *American Psychological Society*, vol. 12, no. 6, pp. 462-6.

Strayer, DL, Drews, FA and Johnson, WA 2003, 'Cell Phone-Induced Failures of Visual Attention During Simulated Driving', *Journal of Experimental Psychology: Applied*, vol. 9, no. 1, pp. 23-32.

Sun Microsystems 2002, *JSR-000082 Java(TM) APIs for Bluetooth(TM) Specification 1.0 Final Release*.

Telematics Research Group 2004, *TRG Europe - Company Presentation*, <http://www.telematicsresearch.de/PDFs/20040812(TRG%20E%20Company%20Presentation)KU.pdf>.

The Legion of The Bouncy Castle *Java Crypto APIs*, viewed Tuesday November 1 2005, <http://www.bouncycastle.org/>.

Tolba, AS, El-Baz, AH and El-Harby, AA 2005, 'Face Recognition: A Literature Review', *Internation Journal of Signal Processing*, vol. 2, no. 1, pp. 88-103.

Trbovich, P and Harbluk, JL 2003, 'Cell phone communication and driver visual behavior: the impact of cognitive distraction', paper presented to CHI '03 extended abstracts on Human factors in computing systems, Ft. Lauderdale, Florida, USA.

Violanti, J 1999, 'Cellular phones and fatal traffic collisions.' *Accid Anal Prev*, vol. 30, pp. 519-24.

Weiser, M 1999, 'The computer for the 21st century', *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 3, no. 3, pp. 3-11.

Wheatley , DJ 2000, 'Beyond the desktop: and into your vehicle', paper presented to CHI '00 extended abstracts on Human factors in computing systems, The Hague, The Netherlands.

WiFi Alliance *WiFi Overview*, viewed Tuesday, 11 October 2005, <http://www.wi-fi.org/OpenSection/why_Wi-Fi.asp?TID=2>.

Williams, J 2002, *Cell Phones and Driving*.

Zhao, Y 2002, 'Telematics: Safe and Fun Driving', *IEEE Intelligent Systems*.

7

7. Appendices

7.1. Appendix A – Password to Key Raw Data (MD5)

This appendix contains raw data relating to the testing of the mobile SNMPv3 USM as described in Section 3.2. Note all figures represent the runtime in seconds.

Nokia 6610	**64b**	**80B**	**800B**	**1KB**	**10KB**	**100KB**	**200KB**	**400KB**	**800KB**	**1MB**
Num Blocks	**1**	**10**	**100**	**128**	**1280**	**12800**	**25600**	**51200**	**102400**	**131072**
1st Run	0.17	0.20	0.49	0.57	4.32	41.98	83.49	166.91	334.09	425.46
2nd Run	0.16	0.19	0.50	0.58	4.30	41.65	83.34	166.68	333.12	426.79
3rd Run	0.16	0.19	0.48	0.58	4.36	41.99	83.73	166.69	334.68	426.56
4th Run	0.16	0.19	0.50	0.57	4.31	41.82	83.18	166.79	333.57	426.12
5th Run	0.16	0.19	0.48	0.58	4.30	41.67	83.27	166.97	333.58	425.64
AV	**0.16**	**0.19**	**0.49**	**0.58**	**4.32**	**41.82**	**83.40**	**166.81**	**333.81**	**426.12**

Nokia 6600	**64b**	**80B**	**800B**	**1KB**	**10KB**	**100KB**	**200KB**	**400KB**	**800KB**	**1MB**
Num Blocks	**1**	**10**	**100**	**128**	**1280**	**12800**	**25600**	**51200**	**102400**	**131072**
1st Run	0.31	0.33	0.44	0.47	1.53	10.38	20.81	40.11	80.22	101.83
2nd Run	0.28	0.28	0.41	0.45	1.44	10.02	19.47	40.16	78.30	101.55
3rd Run	0.27	0.28	0.44	0.48	1.36	9.83	19.89	39.53	80.00	101.44
4th Run	0.31	0.31	0.42	0.44	1.47	10.09	20.31	39.86	78.31	101.94
5th Run	0.28	0.31	0.48	0.45	1.33	10.13	20.14	39.64	79.00	101.16
AV	**0.29**	**0.30**	**0.44**	**0.46**	**1.43**	**10.09**	**20.13**	**39.86**	**79.17**	**101.58**

iMate SP3i	**64b**	**80B**	**800B**	**1KB**	**10KB**	**100KB**	**200KB**	**400KB**	**800KB**	**1MB**
Num Blocks	**1**	**10**	**100**	**128**	**1280**	**12800**	**25600**	**51200**	**102400**	**131072**
1st Run	0.06	0.07	0.08	0.08	0.25	1.77	3.49	6.83	13.46	17.27
2nd Run	0.06	0.06	0.07	0.08	0.32	1.77	3.43	6.41	13.49	17.18
3rd Run	0.06	0.06	0.07	0.08	0.25	1.76	3.42	6.78	13.49	17.26
4th Run	0.06	0.06	0.07	0.08	0.25	1.79	3.45	6.79	13.47	17.25
5th Run	0.06	0.06	0.07	0.08	0.25	1.76	3.43	6.69	13.61	17.48
AV	**0.06**	**0.06**	**0.07**	**0.08**	**0.26**	**1.77**	**3.44**	**6.70**	**13.50**	**17.29**

7.2. Appendix B – Password to Key Raw Data (SHA)

This appendix contains raw data relating to the testing of the mobile SNMPv3 USM as described in Section 3.2. Note all figures represent the runtime in seconds.

Nokia 6610 Nun Blocks	**64b 1**	**80B 10**	**800B 100**	**1KB 128**	**10KB 1280**	**100KB 12800**	**200KB 25600**	**400KB 51200**	**800KB 102400**	**1MB 131072**
1st Run	0.19	0.26	0.91	1.11	9.48	94.02	187.01	373.68	746.44	954.87
2nd Run	0.19	0.26	0.91	1.11	9.46	93.20	186.20	371.71	743.95	952.18
3rd Run	0.19	0.26	0.91	1.11	9.51	93.54	186.45	371.96	746.12	953.66
4th Run	0.19	0.26	0.93	1.12	9.67	93.37	186.00	372.17	744.83	952.82
5th Run	0.19	0.25	0.91	1.13	9.46	93.34	186.92	372.10	743.72	952.26
AV	**0.19**	**0.26**	**0.91**	**1.12**	**9.52**	**93.49**	**186.51**	**372.33**	**745.01**	**953.16**

Nokia 6600 Nun Blocks	**64b 1**	**80B 10**	**800B 100**	**1KB 128**	**10KB 1280**	**100KB 12800**	**200KB 25600**	**400KB 51200**	**800KB 102400**	**1MB 131072**
1st Run	0.30	0.28	0.45	0.47	1.27	9.89	19.48	40.11	79.55	100.02
2nd Run	0.27	0.30	0.44	0.47	1.28	9.75	19.66	38.94	77.47	100.13
3rd Run	0.25	0.27	0.40	0.47	1.24	9.91	19.31	38.69	78.48	101.31
4th Run	0.27	0.28	0.45	0.45	1.37	10.03	19.28	39.52	78.89	101.03
5th Run	0.25	0.25	0.44	0.47	1.28	9.63	19.44	39.17	77.30	100.64
AV	**0.27**	**0.28**	**0.44**	**0.47**	**1.29**	**9.84**	**19.43**	**39.28**	**78.34**	**100.63**

iMate SP3i Num Blocks	**64b 1**	**80B 10**	**800B 100**	**1KB 128**	**10KB 1280**	**100KB 12800**	**200KB 25600**	**400KB 51200**	**800KB 102400**	**1MB 131072**
1st Run	0.06	0.07	0.08	0.09	0.36	2.73	5.81	10.29	21.67	25.17
2nd Run	0.06	0.07	0.08	0.09	0.36	2.74	5.24	10.24	20.26	25.29
3rd Run	0.06	0.07	0.08	0.08	0.38	2.55	5.03	10.23	20.39	27.89
4th Run	0.05	0.06	0.08	0.09	0.34	2.78	5.16	10.25	20.41	27.84
5th Run	0.06	0.06	0.08	0.08	0.33	2.63	5.50	10.28	20.52	25.03
AV	**0.06**	**0.06**	**0.08**	**0.09**	**0.35**	**2.68**	**5.35**	**10.26**	**20.65**	**26.24**

7.3. Appendix C – Face Recognition Test 1 Raw Data

FACE RECOGNITION TEST 1

	1 USER	2 USERS	3 USERS	4 USERS	5 USERS	6 USERS	7 USERS	8 USERS	9 USERS	10 USERS
1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
3	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
4	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
5	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
6	Y	Y	Y	Y	Y	Y	N	Y	Y	Y
7	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
8	Y	Y	Y	Y	Y	Y	Y	Y	Y	N
9	Y	Y	Y	Y	Y	Y	Y	Y	Y	N
10	Y	Y	Y	Y	Y	Y	Y	Y	N	N
11	Y	Y	Y	Y	Y	Y	Y	Y	N	Y
12	Y	Y	Y	Y	Y	Y	Y	Y	N	N
13	Y	Y	Y	Y	Y	Y	Y	Y	N	Y
14	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
15	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
16	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
17	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
18	Y	Y	Y	Y	Y	Y	N	Y	Y	Y
19	Y	Y	Y	Y	Y	Y	Y	N	Y	Y
20	Y	Y	Y	Y	Y	Y	Y	N	Y	Y
21	Y	Y	Y	Y	Y	Y	Y	N	Y	Y
22	Y	Y	Y	Y	Y	Y	Y	N	Y	Y
23	Y	Y	Y	Y	Y	Y	Y	N	Y	N
24	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
25	Y	Y	Y	Y	Y	Y	N	Y	Y	Y
26	Y	Y	Y	Y	Y	N	Y	Y	Y	Y
27	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
28	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
29	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
30	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
31	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
32	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
33	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
34	Y	Y	Y	Y	Y	Y	Y	N	Y	Y
35	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
36	Y	Y	Y	Y	Y	Y	Y	N	N	Y
37	Y	Y	Y	Y	Y	Y	Y	Y	N	Y
38	Y	Y	Y	Y	Y	Y	Y	Y	Y	N
39	Y	Y	Y	Y	Y	Y	Y	Y	Y	N
40	Y	Y	Y	Y	Y	Y	Y	Y	N	N
41	Y	Y	Y	Y	Y	Y	Y	Y	N	Y
42	Y	Y	Y	Y	Y	Y	Y	N	Y	Y
43	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
44	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
45	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
46	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
47	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
48	Y	Y	Y	Y	Y	Y	N	N	Y	Y
49	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
50	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
WRONG	0	0	0	0	0	1	3	9	7	8

7.4. Appendix D – Face Recognition Test 2 Raw Data

FACE RECOGNITION TEST 2

	1 USER	2 USERS	3 USERS	4 USERS	5 USERS	6 USERS	7 USERS	8 USERS	9 USERS	10 USERS
1	Y	N	N	N	N	N	Y	Y	Y	N
2	Y	Y	N	N	N	N	Y	N	N	Y
3	Y	Y	N	N	N	N	N	Y	Y	Y
4	Y	Y	N	N	N	N	N	Y	Y	Y
5	Y	Y	N	Y	N	N	N	Y	Y	Y
6	Y	Y	N	Y	Y	N	N	Y	Y	Y
7	Y	Y	N	Y	Y	N	N	N	Y	Y
8	Y	N	Y	Y	N	N	Y	N	Y	Y
9	Y	N	Y	Y	N	N	Y	N	Y	Y
10	Y	N	N	Y	Y	N	Y	Y	Y	Y
11	Y	N	N	N	N	N	Y	Y	N	Y
12	Y	N	N	Y	N	N	Y	Y	Y	Y
13	Y	N	N	Y	Y	N	Y	Y	Y	N
14	Y	N	N	N	Y	N	N	Y	Y	N
15	Y	N	Y	N	Y	N	N	N	Y	Y
16	Y	N	Y	Y	N	N	Y	N	N	Y
17	Y	N	N	Y	N	N	N	Y	N	N
18	Y	N	Y	Y	N	N	N	Y	Y	N
19	Y	Y	Y	N	N	N	N	Y	Y	Y
20	Y	Y	Y	Y	Y	N	N	N	Y	Y
21	Y	Y	Y	Y	Y	N	Y	N	Y	Y
22	Y	Y	Y	N	N	N	Y	Y	N	Y
23	Y	Y	N	Y	N	N	N	Y	N	N
24	Y	Y	N	Y	Y	N	N	Y	N	N
25	Y	Y	N	N	Y	N	N	Y	Y	N
26	Y	Y	N	N	N	N	N	Y	Y	N
27	Y	Y	N	N	N	N	N	Y	Y	Y
28	Y	Y	Y	Y	N	N	Y	N	Y	N
29	Y	N	Y	Y	N	N	Y	Y	Y	Y
30	Y	N	N	N	Y	N	N	Y	Y	Y
31	Y	N	Y	N	Y	N	N	Y	N	Y
32	Y	N	N	N	N	N	N	N	N	Y
33	Y	Y	N	N	N	N	Y	N	N	Y
34	Y	Y	N	N	Y	N	N	N	N	Y
35	Y	Y	N	Y	Y	Y	N	N	Y	N
36	Y	Y	N	N	Y	Y	N	Y	Y	N
37	Y	Y	N	Y	Y	Y	Y	Y	Y	N
38	Y	N	N	Y	Y	Y	Y	Y	Y	Y
39	Y	N	N	Y	Y	Y	N	Y	N	Y
40	Y	N	N	Y	Y	N	N	Y	N	Y
41	Y	N	N	Y	Y	N	N	Y	N	Y
42	Y	N	N	Y	Y	N	N	Y	N	Y
43	Y	Y	N	Y	N	N	Y	Y	N	Y
44	Y	Y	N	Y	Y	Y	Y	Y	N	Y
45	Y	Y	N	Y	Y	Y	N	Y	Y	Y
46	Y	Y	N	Y	N	N	N	Y	N	Y
47	Y	Y	N	N	N	N	N	Y	N	Y
48	Y	Y	N	N	N	N	Y	Y	Y	Y
49	Y	Y	N	N	Y	N	N	Y	N	Y
50	Y	N	N	Y	Y	Y	Y	Y	N	Y
WRONG	0	22	38	21	25	42	30	13	21	13

7.5. Appendix E – Face Recognition Test 3 Raw Data

FACE RECOGNITION TEST 3

	1 USER	2 USERS	3 USERS	4 USERS	5 USERS	6 USERS	7 USERS	8 USERS	9 USERS	10 USERS
1	Y	N	N	N	N	N	N	N	N	N
2	Y	N	N	N	N	N	N	N	N	N
3	Y	N	N	N	N	N	N	N	N	N
4	Y	N	N	N	N	N	N	N	N	N
5	Y	N	N	N	N	N	N	N	N	N
6	Y	N	N	N	N	N	N	N	N	N
7	Y	Y	N	N	N	N	N	N	N	N
8	Y	N	N	N	N	N	N	N	N	N
9	Y	N	N	N	N	N	N	N	N	N
10	Y	Y	N	N	N	N	N	N	N	N
11	Y	N	N	N	N	N	N	N	N	N
12	Y	N	N	Y	N	N	N	Y	N	Y
13	Y	N	N	N	N	N	N	N	N	Y
14	Y	N	N	N	N	N	N	N	N	N
15	Y	N	N	N	N	N	N	N	N	N
16	Y	N	N	N	N	N	N	N	N	N
17	Y	N	Y	N	N	N	N	N	N	N
18	Y	N	N	N	N	N	N	N	N	N
19	Y	N	N	Y	N	N	N	N	N	N
20	Y	N	N	Y	N	N	N	N	N	N
21	Y	N	N	N	N	N	N	N	N	N
22	Y	N	N	N	N	N	N	N	N	N
23	Y	N	N	N	N	N	N	N	N	N
24	Y	N	N	N	N	N	N	N	N	N
25	Y	N	Y	N	N	N	Y	N	N	N
26	Y	N	N	N	N	N	N	N	N	N
27	Y	N	N	N	N	N	N	N	N	N
28	Y	N	N	N	N	N	N	N	N	N
29	Y	N	N	Y	N	Y	N	N	N	N
30	Y	N	N	Y	N	Y	N	N	N	N
31	Y	N	N	Y	N	N	N	N	N	N
32	Y	N	N	N	N	N	N	N	N	N
33	Y	N	N	N	N	Y	N	N	N	N
34	Y	N	N	N	N	N	N	N	Y	N
35	Y	N	N	N	N	N	N	N	Y	N
36	Y	N	N	N	N	N	N	N	N	N
37	Y	N	N	N	N	N	N	N	Y	N
38	Y	N	N	N	N	N	N	N	N	N
39	Y	N	N	N	N	N	N	N	N	N
40	Y	N	N	N	N	N	N	N	N	N
41	Y	N	N	N	N	N	N	N	N	N
42	Y	N	N	N	N	N	N	N	N	N
43	Y	N	N	Y	N	N	N	N	N	N
44	Y	N	Y	N	N	N	N	N	N	N
45	Y	N	N	N	N	N	Y	N	N	N
46	Y	N	Y	N	N	N	N	N	N	N
47	Y	N	N	N	N	N	N	N	N	N
48	Y	N	N	N	N	N	N	N	N	N
49	Y	N	N	N	N	N	N	N	N	N
50	Y	N	N	N	N	N	N	N	N	N
WRONG	0	48	46	43	50	47	48	49	47	48